

IS 4·AMIGA·PC·C64·C128·C16 PLUS 4·AMIGA·

YOUR

AUGUST 1988 \$7.95

# COMMODORE



*Mac images on Amiga*

## REVIEWED

On The PC: 386 Attack Sub, Abraham's Battle Tank, Battle Hawks 1942

For The C64: Pinball Fantasies, Speedball, Navy Mines, Star Trek

## UNBEATABLE PROGRAMS

Address Book, Disk Scrubber, Thesaurus, Wordline and more

ISSN 0269-8297



# DATTEL ELECTRONICS



## 256K SUPEROM EXPANDER

- 1 Tested thoroughly to meet quality control standards
- 1 Program your own EPROMs using our EPROM programmer
- 1 No need to learn lots of complicated just make a cartridge from the program card
- 1 Fully secure address via power up
- 1 Built-in write enable software controls
- 1 On-board EPROM programmer features will take your own programs - built-in write enable software built-in software (EPROM) programmer
- 1 Accepts 27160, 27160, 27160, 27160, 27160, 27160
- 1 On-board microprocessor card
- 1 On-board microprocessor card - no program to load

**ONLY £29.99**



## RESET CARTRIDGE

- 1 Microprocessor reset button
- 1 AUTO reset (one press) type reset
- 1 Software controlling system can reset microprocessor
- 1 Programmable reset button
- 1 100% power down, no damage to data
- 1 100% power down, no damage to data
- 1 100% power down, no damage to data
- 1 100% power down, no damage to data
- 1 100% power down, no damage to data
- 1 100% power down, no damage to data

**ONLY £5.99**



## 3 SLOT MOTHERBOARD

- 1 Accepts 3 cartridges
- 1 Prints full test results
- 1 On-board safety test
- 1 Fully featured
- 1 High power PPS
- 1 Extra features

**ONLY £16.99**



## MIDI G4

FULL SIZE INTERFACE FOR THE G4/10 AT A BARGAIN PRICE

- 1 MIDI interface for G4/10 AT
- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT

**ONLY £29.99**



## DIGITAL SOUND SAMPLER

- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT

**ONLY £49.99**

## COM-DRUM

COM-DRUM EDITOR

- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT

**ONLY £5.99**

ONLY £5.99

## COM-DRUM EDITOR

- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT

**ONLY £4.99 ONLY**

## DATA RECORDER

- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT

**ONLY £24.99**



## DEEP SCAN BURST NIBBLER

- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT

**ONLY £24.99 COMPLETE**

SOFTWARE ONLY £12.99

CABLE ONLY £14.99

DATTEL ELECTRONICS Ltd. 100, 101, 102, 103, 104, 105, 106, 107, 108, 109, 110, 111, 112, 113, 114, 115, 116, 117, 118, 119, 120, 121, 122, 123, 124, 125, 126, 127, 128, 129, 130, 131, 132, 133, 134, 135, 136, 137, 138, 139, 140, 141, 142, 143, 144, 145, 146, 147, 148, 149, 150, 151, 152, 153, 154, 155, 156, 157, 158, 159, 160, 161, 162, 163, 164, 165, 166, 167, 168, 169, 170, 171, 172, 173, 174, 175, 176, 177, 178, 179, 180, 181, 182, 183, 184, 185, 186, 187, 188, 189, 190, 191, 192, 193, 194, 195, 196, 197, 198, 199, 200, 201, 202, 203, 204, 205, 206, 207, 208, 209, 210, 211, 212, 213, 214, 215, 216, 217, 218, 219, 220, 221, 222, 223, 224, 225, 226, 227, 228, 229, 230, 231, 232, 233, 234, 235, 236, 237, 238, 239, 240, 241, 242, 243, 244, 245, 246, 247, 248, 249, 250, 251, 252, 253, 254, 255, 256, 257, 258, 259, 260, 261, 262, 263, 264, 265, 266, 267, 268, 269, 270, 271, 272, 273, 274, 275, 276, 277, 278, 279, 280, 281, 282, 283, 284, 285, 286, 287, 288, 289, 290, 291, 292, 293, 294, 295, 296, 297, 298, 299, 300, 301, 302, 303, 304, 305, 306, 307, 308, 309, 310, 311, 312, 313, 314, 315, 316, 317, 318, 319, 320, 321, 322, 323, 324, 325, 326, 327, 328, 329, 330, 331, 332, 333, 334, 335, 336, 337, 338, 339, 340, 341, 342, 343, 344, 345, 346, 347, 348, 349, 350, 351, 352, 353, 354, 355, 356, 357, 358, 359, 360, 361, 362, 363, 364, 365, 366, 367, 368, 369, 370, 371, 372, 373, 374, 375, 376, 377, 378, 379, 380, 381, 382, 383, 384, 385, 386, 387, 388, 389, 390, 391, 392, 393, 394, 395, 396, 397, 398, 399, 400, 401, 402, 403, 404, 405, 406, 407, 408, 409, 410, 411, 412, 413, 414, 415, 416, 417, 418, 419, 420, 421, 422, 423, 424, 425, 426, 427, 428, 429, 430, 431, 432, 433, 434, 435, 436, 437, 438, 439, 440, 441, 442, 443, 444, 445, 446, 447, 448, 449, 450, 451, 452, 453, 454, 455, 456, 457, 458, 459, 460, 461, 462, 463, 464, 465, 466, 467, 468, 469, 470, 471, 472, 473, 474, 475, 476, 477, 478, 479, 480, 481, 482, 483, 484, 485, 486, 487, 488, 489, 490, 491, 492, 493, 494, 495, 496, 497, 498, 499, 500, 501, 502, 503, 504, 505, 506, 507, 508, 509, 510, 511, 512, 513, 514, 515, 516, 517, 518, 519, 520, 521, 522, 523, 524, 525, 526, 527, 528, 529, 530, 531, 532, 533, 534, 535, 536, 537, 538, 539, 540, 541, 542, 543, 544, 545, 546, 547, 548, 549, 550, 551, 552, 553, 554, 555, 556, 557, 558, 559, 560, 561, 562, 563, 564, 565, 566, 567, 568, 569, 570, 571, 572, 573, 574, 575, 576, 577, 578, 579, 580, 581, 582, 583, 584, 585, 586, 587, 588, 589, 590, 591, 592, 593, 594, 595, 596, 597, 598, 599, 600, 601, 602, 603, 604, 605, 606, 607, 608, 609, 610, 611, 612, 613, 614, 615, 616, 617, 618, 619, 620, 621, 622, 623, 624, 625, 626, 627, 628, 629, 630, 631, 632, 633, 634, 635, 636, 637, 638, 639, 640, 641, 642, 643, 644, 645, 646, 647, 648, 649, 650, 651, 652, 653, 654, 655, 656, 657, 658, 659, 660, 661, 662, 663, 664, 665, 666, 667, 668, 669, 670, 671, 672, 673, 674, 675, 676, 677, 678, 679, 680, 681, 682, 683, 684, 685, 686, 687, 688, 689, 690, 691, 692, 693, 694, 695, 696, 697, 698, 699, 700, 701, 702, 703, 704, 705, 706, 707, 708, 709, 710, 711, 712, 713, 714, 715, 716, 717, 718, 719, 720, 721, 722, 723, 724, 725, 726, 727, 728, 729, 730, 731, 732, 733, 734, 735, 736, 737, 738, 739, 740, 741, 742, 743, 744, 745, 746, 747, 748, 749, 750, 751, 752, 753, 754, 755, 756, 757, 758, 759, 760, 761, 762, 763, 764, 765, 766, 767, 768, 769, 770, 771, 772, 773, 774, 775, 776, 777, 778, 779, 780, 781, 782, 783, 784, 785, 786, 787, 788, 789, 790, 791, 792, 793, 794, 795, 796, 797, 798, 799, 800, 801, 802, 803, 804, 805, 806, 807, 808, 809, 810, 811, 812, 813, 814, 815, 816, 817, 818, 819, 820, 821, 822, 823, 824, 825, 826, 827, 828, 829, 830, 831, 832, 833, 834, 835, 836, 837, 838, 839, 840, 841, 842, 843, 844, 845, 846, 847, 848, 849, 850, 851, 852, 853, 854, 855, 856, 857, 858, 859, 860, 861, 862, 863, 864, 865, 866, 867, 868, 869, 870, 871, 872, 873, 874, 875, 876, 877, 878, 879, 880, 881, 882, 883, 884, 885, 886, 887, 888, 889, 890, 891, 892, 893, 894, 895, 896, 897, 898, 899, 900, 901, 902, 903, 904, 905, 906, 907, 908, 909, 910, 911, 912, 913, 914, 915, 916, 917, 918, 919, 920, 921, 922, 923, 924, 925, 926, 927, 928, 929, 930, 931, 932, 933, 934, 935, 936, 937, 938, 939, 940, 941, 942, 943, 944, 945, 946, 947, 948, 949, 950, 951, 952, 953, 954, 955, 956, 957, 958, 959, 960, 961, 962, 963, 964, 965, 966, 967, 968, 969, 970, 971, 972, 973, 974, 975, 976, 977, 978, 979, 980, 981, 982, 983, 984, 985, 986, 987, 988, 989, 990, 991, 992, 993, 994, 995, 996, 997, 998, 999, 1000.



## ROBOTER 64

ROBOTER 64

- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT

**ONLY £39.99**

ONLY £39.99

ONLY £39.99



## TOOLKIT IV

THE ULTIMATE DISK TOOLKIT FOR THE 1040/1041

- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT
- 1 Full size interface for G4/10 AT

**ONLY £5.99**

# DATTEL ELECTRONICS



## TURBO ROM II

### NEW CLONEMASTER

- ☐ This great software helps to save time by saving valuable machine specific features of your system ready to be loaded
- ☐ The same for settings needed to set up
- ☐ The latest VGA, VGA cloning instantly copies the program to make it perfect aligned to the correct monitor (producing a top picture from the original in many cases)
- ☐ A full auto-format to avoid machine type cloning
- ☐ Works with almost any program including most business software
- ☐ Even more internal type features. Program cannot be fully copied completely the same machine
- ☐ Simply press 'Play' and it will transfer a game. (Based on the other data in the other)
- ☐ This can even make a backup with one file for the program
- ☐ This is a total feature machine no program to load the results are amazing!

ONLY £9.99

**TURBO ROM II IS A REPLACEMENT FOR THE ACTUAL ROM INSIDE YOUR 84. IT PROVIDES SUPERFAST LOAD/SAVE ROUTINES.**

- ☐ Small load/save program of 2-4 bytes normal speed
- ☐ Saving of 2-4 bytes normal speed on disk
- ☐ Enhanced 200K support (including 100K) format
- ☐ Programmed loading (copy load strategy) 100K disk
- ☐ Software to normal format to a disk of 100K
- ☐ Program automatic file copy
- ☐ Packed/unpacked 100K/200K
- ☐ How file name
- ☐ Format to disk drive (no software usually required) (the same for a 512K disk ROM) (see how to be downloaded)

ONLY £14.99



### BLUE CHIP DRIVE



- ☐ Excellent backup supply for most computers
- ☐ Three routes to 100K double sided disks (all made)
- ☐ Fully 100K/100K compatible
- ☐ It is also a very portable disk and computer drive available for the Commodore disk on disk drive Commodore disk (100K)
- ☐ Improved compatibility with all disks on most file type
- ☐ At least a top quality drive at a reasonable price

**ONLY £139.99**  
ADD £5.00 FOR CARRIER DELIVERY

### ULTRA CRUNCHER

- ☐ The ultimate cartridge based program compressor
- ☐ Compresses to up to 50%
- ☐ It compresses programs on the cartridge
- ☐ From loading/running programs
- ☐ Full 200K support including full format

**ONLY £12.99**  
on cartridges

Available to ship only for 2048 Superdisk based for

**ONLY £7.99**

### FAST HACK'EM

- ☐ Most common disk drivers - all on the one disk
- ☐ Single 100K/100K - Copy on entire disk to just 100K
- ☐ Copy last file Copy - Apply copy on entire file to entire format
- ☐ No other product is such a good 100K/100K on the loading/running file

**ONLY £9.99**



### SMART CART

- ☐ 1000 pages ROM
- ☐ On-line format down to 4 pages
- ☐ Simply load the program, you require data from the outside. The cartridge can then be removed just like a ROM cartridge
- ☐ Under your own cartridge containing software from additional format
- ☐ 400K format - 400K pages
- ☐ Three cartridges with 100K/100K and full format (included)
- ☐ 400K to 100K for special programming techniques

**ONLY £29.99**

### RAM DISK

- ☐ Three page format that fits in 100K RAM disk
- ☐ 100K internal storage area for the program
- ☐ 100K/100K compatible load, save, transfer, protect
- ☐ Program data to internal when program is running up
- ☐ Full removal of disk contents

**ONLY £9.99**

### BLAZING PADDOLE

**A COMPLETE LIGHTPEN/GRAPHICS ILLUSTRATOR PACKAGE.**

- ☐ A full free three graphics package of a system which should not leave much out
- ☐ Complete with a full optical lightpen system for pen point accuracy
- ☐ Pen and pen to use the system. (Other display a system with a drawing tablet)
- ☐ Blazing Paddole will also work with other color light pens including 'reusable' blue lightpen tablets available too
- ☐ Multi-levels and menu handling
  - Range of brush
  - Arrows
  - Rectangle
  - Circle
  - Ellipse
  - Line
  - Fill
  - Pattern fill
  - Text
  - Advanced screen editing - zoom
- ☐ 100K format
- ☐ On a page which displays a screen light pen to be used to draw a picture

**ONLY £24.99**

COMPLETE BLAZING PADDLE SYSTEM

BLAZING PADDLE ONLY  
**£12.99**

**ALL ORDERS NORMALLY DESPATCHED WITHIN 48 HRS**

## HOW TO ORDER ...

**BY PHONE**



0744 744747

order line

Card Line

**BY POST**



Send money, cheque, postal order to

order line

Card Line

**FAX**

0744 744747

order line

Card Line

PHONE AND TELEPHONE CREDIT CARD OR PAY BY PHONE

AND PAYMENT BY CREDIT CARD (VISA/MASTERCARD)

CARD PAYMENTS - Please contact your bank for card payment (1-24)



**DATTEL ELECTRONICS LTD PENTON INDUSTRIAL ESTATE**

**GOVAN ROAD, PENTON, WOLVERHAMPTON, ENGLAND**

**SALES ONLY**

0744 744747

**COMMERCIAL ONLY**

0744 744747

# CONTENTS

VOLUME 5  
NUMBER 11



PCP Studios



Warner Bros.



Warner Bros.

## FEATURES

- |   |    |   |    |
|---|----|---|----|
| • <b>Foreground</b><br>All you ever wanted to know about the Plus/4         | 6  | • <b>Star Trek</b><br>We come in peace, shoot to kill...  | 44 |
| • <b>Autocode</b><br>An ambitious new pathfinding system for motorists      | 18 | • <b>American Club Sports</b><br>Indoor games, american style                                     | 49 |
| • <b>Battletech</b><br>A massive new role-playing game                      | 33 | • <b>Speedball</b><br>Definitely not for ladies, this one   | 50 |
| • <b>MacPics on Amiga</b><br>How to port Macintosh graphics onto your Amiga | 34 | • <b>Circus Attractions</b><br>Falling off horses, throwing lemons at women, and much, much more! | 51 |
| • <b>Navy Moves</b><br>A tough new tactical challenge                       | 41 | • <b>Geos 128</b><br>The latest addition to the Desktop Environment Series for the 128            | 52 |
| • <b>PC Games</b><br>The latest games on the Commodore PC                   | 42 | • <b>Project Firestart</b><br>Hearse in deepest space   | 76 |

## REGULARS

- |   |    |   |    |
|---|----|---|----|
| • <b>Data Statements</b><br>More news from the world of Commodore           | 8  | • <b>128 Corner</b><br>Comments and queries from 128 users          | 68 |
| • <b>Extending Basic</b><br>How to add a trace routine to Commodore Basic   | 14 | • <b>Software for Sale</b><br>If you don't want to type it, buy it! | 70 |
| • <b>PC Corner</b><br>The first of a regular series for Commodore PC owners | 20 | • <b>Mailbag</b><br>Your chance to air your views                   | 78 |
| • <b>Amiga Workbench</b><br>Helpful hints and tips for Amiga users          | 40 | • <b>Backpage</b><br>Corrections, errata and 'The Nibbles'          | 82 |

## PROGRAMS

- |   |    |  |    |
|---|----|--|----|
| • <b>Reasoning on the 128</b><br>Turn your humble 128 into an expert system | 22 | • <b>Disk Scrambler</b><br>Protect your disks from prying eyes                       | 54 |
| • <b>Thesaurus</b><br>Stuck for words? This program can help you            | 28 | • <b>Machine Code Disk Programming</b><br>The intricacies of Machine Code unravelled | 56 |
| • <b>Windows on a Maze</b><br>A maze game for beginners                     | 37 | • <b>Inside the 1541</b><br>Pascal Moore pokes around inside the 1541 disk drive     | 60 |
| • <b>Address Book</b><br>A database for 128 users                           | 45 | • <b>Memory Management</b><br>Explore the intricacies of your 128's memory           | 62 |

# FourGround!

*Plus/4 Past and Present •*

*By Mark Everingham*

**1**984 — the year of the first microcomputer wars. C&G Lawson took four gold medals in the Summer Olympics, Panasonic took two the Nobel Peace Prize, and Commodore introduced the Plus/4 computer. All right, maybe the advent of the Plus/4 did not have quite the same impact, but its arrival did create something of a stir in the microcomputer world. The magazine *Your Computer* described the machine as a "QOL-Master" and suggested that the Plus/4 would replace the BBC Micro at school and business because of the powerful CBM BASIC V2.5 and the integrated business software. Unfortunately, those nebulous claims were not to be fulfilled. In recent times the Plus/4 has lost the support of most software houses, and has sadly slipped into relative obscurity. Even so, the Plus/4 did have considerable, if short-lived success in both the UK and, perhaps more importantly, in Germany and the USA. I was convinced myself of how many people were still using the machine by response to a bag which appeared in a program of mine published in *Your Commodore* last year. I received numerous letters from as far as the States and Holland, and was surprised by how many people had actually had a go at solving the bag, rather than just writing for help. Although largely forgotten by the software industry, the Plus/4 is still far from dead, five years after its conception, and now it has finally received the recognition it deserved, in this form — an area of the magazine set aside solely for the Plus/4 computer, and the name — **FourGround!**

## Aims of The Series

When I was approached by *Your Commodore* to write the series on the Plus/4 I accepted immediately — the Plus/4 is simply the most friendly, usable machine I have ever come across and has been disregarded for far too long. However, when I started to think of some ideas to write about, I realised the enormous job I had taken on. The fact is that the Plus/4 excels in so many things, and has so many different uses from business to games

playing that it is hard to target any one area of interest. If I confined myself to mid-level language programming I may alienate the Basic-only programmers, and of course I mustn't forget those who use their Plus/4 mainly for playing games. I have therefore selected a number of aspects of the Plus/4 which I consider to be definitive of the computer, and tried to cater for as many diverse interests as possible. If demand is high enough, I would also be happy to run a "Plus/4 Problems" type section. If you have any ideas, suggestions, questions or requests, please write to me at the address shown at the end of this article. If you require a reply, please do include an S.A.R.

Having got the introductions over, I thought that this month rather than leaping straight into a new topic, we'd take a look at what the Plus/4 offers. What follows is a brief overview of the Plus/4 system, and an inventory of those features of the Plus/4 which will be covered in this space in future issues of the magazine. Because the Plus/4 has so much to offer, the less obvious aspects of the computer are often forgotten. This month's overview should act as a memory refresher and a starter of subjects for future coverage.

## An Overview Of The Plus/4 Computer

The Plus/4 computer has essentially three modes of operation — running Basic programs, running the built-in business software, and running software written in the user's own machine-code programs. The default of Basic provided with the machine is not the CBM's old Basic V2.0 which goes back to CBM PET and VIC-20 days, but is a far improved version 3.5. The main difference is the provision of commands to handle graphics and sound, which had to be performed using tedious strings of POKEs on the Plus/4's big brother, the CBM. In addition, it is possible to write far more elegant and structured programs in Basic 3.5 because of the **GO...LOOP** constructs used in the earlier versions of Commodore Basic. The set of graphics commands included are so complete a collection as can be found on any computer. Basic 3.5 also offers comprehensive "house-keeping" facilities

like program numbering and automatic line numbering which along with a set of eight redefinable function keys make programming in Basic simplicity itself. Commodore's enhanced full-screen editor found on the Plus/4 means that prototyping of screen displays is easy, and program editing fast and reliable. Although Plus/4 Basic is not the speediest language known, its importance and usefulness should not be underestimated. By its use of ROM/RAM paging, Basic can use the full 64k RAM of the Plus/4 computer. However, arguably the most useful aspect of Basic 3.5 is that it offers unique features which allow the machine-code programmer to easily add new commands to the Basic language. This means that there should never be any need to use unfriendly POKEs and SYS calls. Unfortunately, like so many of the Plus/4's other features, the extension of Basic does not seem to be documented in any books or manuals, so you can expect a full exposure in the coming months.

The second major mode of Plus/4 operation is the built-in "3+1" business software: the integrated word-processor, spreadsheet, database and graph software. Two often the aspect of the Plus/4 is totally disregarded, yet while the software is a bit spartan, it is usable and with a little ingenuity it can be made to perform a wide range of extra functions due to Commodore's generous use of RAM-memory. I shall be proceeding in *FourGround* program which reviews the lack of word-count, copy quantity when printing, apostrophe conversion as well as providing more complex functions such as central-code handling and facilities for the insertion of screen pictures into documents, just like the graphical word-processors available on the Amiga.

By far the most interesting and powerful aspect of the Plus/4 is machine-code, otherwise known as *Machine Language* or *Assembly Language*. Because of the Plus/4's built-in machine-code monitor TROMON, the Plus/4 is the perfect machine on which to learn machine language. The TROM Processor in the Plus/4 is a way to learn and powerful in its simplicity. The inclusion of a Kernel ROM in the Plus/4 makes device handling, disk-drive access and printer control easy to grasp. Machine language registra-

My hat is certainly tilted to it, having gained the reputation of a language beyond the reach of casual mortals. This is not the case — machine-code can be simple and rewarding to use. The problem is usually a bad method of teaching, as I shall be presenting a readable, easily understood introduction to this fascinating aspect of Plus/4 programming some time in the future.

Of course, possibly the most important aspect of Plus/4 machine-code is correct use of the TED chip which replaces the VIC chips found in the older C64 and VIC60 machines. The TED chip is a complex graphics/scroll handling chip which offers a host of features.

High-resolution Text Multi-colour Text and Extended Mode screen displays (32 colours and addressable character-sets of either 128 or 256 characters, Hardware controlled flicking and text reversal), High resolution and Multi-colour graphics modes with facilities for split-screen operation, Hardware smooth scrolling in both text and graphics modes, Scatter (Scrambled Position) interrupts, Faster interrupts at speeds of up to 1MHz and three hardware timers — one reloadable, Full ROM/RAM paging and facilities for paging internal VIC.

Fastway ROM chips, Two channel sound output and white noise generator. The list could go on far ever. There are simply so many features that while most people understand the basic principles, they do not appreciate just how versatile even the simplest function can be when used in unorthodox ways. As an example, try entering the listing (On running the program, a set of concentric rings will be drawn and then the screen should begin to flash wildly. Now depress the [SHIFT-LOCK] key.

Not something you'd normally do when running a program. Immediately you should see the screen resolve into a pattern of colours gliding effortlessly

10	COLOR 0:COLOR 4,1:0
	COLOR 1,1:0
20	GRAPHIC 1,1 (P)
30	FOR K=0 TO 50 STEP 10
40	CIRCLE 1,100,100,R P-P
50	IF P=1 THEN PAINT
	1000,K1-R
60	NEXT R
70	COLOR 0,1:0
80	GRAPHIC 1 GRAPHIC 3
90	GOTO 30

up the background of the screen, amazing when you consider that the whole special effect is being produced by just three basic commands and not a speck of machine-code at all! So how does the program work? If you stop the program and type GRAPHIC 1 you'll see a pattern of black and white rings, but in GRAPHIC 3, the same pattern becomes different colours in the Multi-colour mode. Normally, switching between the two screens just causes the display to flicker. Depressing the [SHIFT-LOCK] key, or holding down any other key makes the Plus/4's interrupts run a bit slower as the keyboard is scanned. This in turn brings the speed of flicker down, so a semi-slow to the TV picture updating speed, resulting in the rolling colours effect. OK, so it's really just a gimmick, but from the obscure use of a simple function, we've learnt something about graphics modes, keyboard scanning,

interrupt handling and TV Picture Handling (or the ubiquitous Ramay). It is often unconventional title programs like this which teach you the most. So, if you have any interesting short programs, send them on!

To complement such Plus/4-specific subjects, I should like to include features on computing previously not tried on the Plus/4, but fully within its capabilities. Simple artificial intelligence and strategy programming for example. Another subject that you can be sure of coverage in the near future is the magnificent art and the whole field of fractal and chaotic maths. The exhilarating new area of computing has usually been confined to the Amiga and PC's of the world but some beautiful effects can be achieved with incredibly simple basic programs on the Plus/4, with its 121 colours adding a whole new dimension to the subject.

The Plus/4, as I have tried to demonstrate by this brief overview has a wide scope of ability for any attempt. I hope I have whetted your appetite for things to come. Next month we'll kick off with something a bit more substantial. Remember, the section of the magazine should be for you the Plus/4 Owners, so please do write to the address for any letters to —

**FOREGROUND**  
27 Collingwood Road,  
Redland,  
Bristol BS6 6PD



**Editor:** Angus Corder  
**Design:** Robin  
**Paul Wilkinson**  
**Technical Editor:** Paul Bee  
**Group Editor:**  
 Mark Webb  
**Advertisement Manager:**  
 Paul Kenyon  
**Ad Copy Control:**  
 Ruth Taylor  
**Artist:** Alan Bain  
**Designer:** Mark Newman  
**Origination:** John  
**Printers:** Clive Webb

Your Commodore magazine from 64 is a monthly magazine appearing on the first Friday of each month. Angus Specialist Publications Limited, Editorial and Administration office, Four Commodore, Regus House, Royston, Herts SG8 5JH. Telephone: 0462-6601. Indusystem rules apply. Application to Your Commodore Subscriptions Department, Inform Ltd, 3 River Park, Luton - Bedfordshire LU1 3HL. U.S.A. Subscriptions Agent: Wile Data Worldwide, P.O. Box 990, 4014 West 12th Street, Torrance, CA 90501 U.S.A.

**ARGUS  
 PRESS  
 GROUP**

The content of this publication including all articles, designs, drawings and programs and all copyrights and other intellectual property rights therein belong to Argus Specialist Publications Limited. All rights reserved by the Law of Copyright and other intellectual property rights and by virtue of patent and copyright international are specifically reserved to Argus Specialist Publications Limited and any reproduction requires the prior written permission of the Company. © 1989 Distribution of this publication is by subscription only. London SW16 4JG. Printed in Great Britain. Physicist Optics are the opinion of the reviewers and not necessarily those of the magazine. While every effort is made to thoroughly check programs published for errors, we cannot be held responsible for any errors that do occur.

ISSN 0260-6277

**ABC**  
OF COMPUTERS

## Data Statements

### PD For All

Some of the best software available for the Commodore range of computers is available in the Public Domain. This is software that is freely distributable, as long as only a minimal charge is made for the disk and handling.

Kingsway Computer Services has informed us of the availability of its free PD catalogues, showing just what software the company is offering. Programs are available on disk only, and cover a large range of subjects such as Education, Utilities, Home Accounts and Games. PD disks are available for the Plus4, C64 and C65.

For your FREE Kingsway PD catalogue either write or telephone Kingsway Computer Services at 146 Renshaw Road, Sheffield, S8 9QE. Tel: (0142) 588429.

### Over the Top?

De Cole Marking are launching four new joysticks, including a top-of-the-range stick that briefly goes where no joystick has gone before by giving the games player not only arc fire buttons, and an analogue with speed control, but also a built-in digital stopwatch. The three joysticks range in price from £8.95 to £19.95.



Yes, readers - a quarter of the Cole joysticks are going for a euro.

### For Book Worms

Probably the best known UK publisher of Electronics and Computer books in the UK is Bernard Babson. A selection of the company's massive range of small paperback books can be found in most good book shops. Well, you may like to know that you can now obtain a free copy of their 1989 catalogue giving information on all of their titles, ranging from *Secrets of the Commodore 64* to the *Microcom Calculator Day's Handbook*.

For your free catalogue write to Bernard Babson (publishing) Ltd, The Grampians, Stephenson Park Road, London W6 7NF. Tel: 01-863 2581.

### New Names for Imageworks

Authors Taghore and Pete Lemm, better known as Starlight Software, authors of the 8-bit games *Real L.E.D.* and *Deadlines*, have found a new home with Imageworks. Allied with the firm's first 16-bit game, should be out any time now on the Amiga, and a C64 version will soon follow.

*Deadlines* is a fantasy role-playing game, but the horizontally split screen which offers a new planet mode adds a new dimension to the genre of games.

### Cheaper Lasers

Being a laser printer is normally a big expensive business, but Qume has just launched a unique upgrading scheme to ease the purchase of their CrystalPoint WP and Series III printers. The user can buy a CrystalPoint



WP at £999 for basic text editing, and upgrade it to a series II when text and graphics are required. The series II (£1,495) and the WP can be modified to the top of the range Publisher PostScript language compatible, which retails at £2,999.

### Bank Desks

**N**US find access men to have found a solution to overcrowded offices with the introduction of a new two tier workstation. The two tier system is designed for use in computer rooms where space is at a premium, and there are a large number of terminals around.



Elaine and Anne, authors of *Blind by Night*



Star's new range of very competitively priced 24 pin printers

### Saving Stars On Site

**I**f you require high quality output from your dot matrix printer, then you really need to get hold of a 24 or 48 pin printer. Now Star are going to bet up the market by offering a range of 24 pin printers with an output quality that's equal to that of a 48 pin printer. The company is also offering 12 months on-site parts and labour warranty. That's usually, a 48 pin printer could cost you in excess of £2,000. Now Star has brought prices for 48 pin quality to below £600.

Prices for the new printers are £590 for the 10 inch X824-10, and £710 for the 15 inch X824-15. Both these printers have a Super Letter Quality mode offering a character matrix of 48 x 35 dots, the equivalent resolution of cheap 48 pin printers. A range of 17 fonts are supplied as standard with the printers, and more can be purchased as you require them. A colour option is also available for both printers.

For more information contact Star at Green House, 48 Highbury Road, London N5 2BS. Tel 01-546 1000.

### New Fonts For Laser

**I**f you own a HP or IBM laser printer for use with your computer, then the price of new font cartridges has probably put you off buying new fonts for the printer. Now C Tech is offering a new range of font cartridges that are compatible with IBM and HP laser printers. They cost around the same price as the "real" cartridges, but offer twice as many options. C Tech are also willing to make quotes for people who want their own fonts on cartridge.

The font library consists of 12 cartridges, all already loaded for use with C Tech's C-85 printer and scanner. Once a pattern or design has been scanned or merged with text, the whole effect can be maintained by using connecting fonts from C Tech's library.



C Tech's range of IBM and HP compatible font cartridges - from as many options

[illegible]

Age Group	Percentage of Respondents
18-29	85%
30-39	80%
40-49	75%
50-59	70%
60-69	65%
70-79	65%
80+	70%

**F**urniture and playing games are options for keeping your child occupied. But the value of a dog, which is a fully the total benefits to a home. It adds to which adds two powers to make the process more enjoyable. The same 10 months. The role in apartment is a change with the owner to share and the baby is more and more. And the dog has been created.

that the system under review has been reviewed with regard to the following issues:

As a result, the  $\beta$ -phase is not observed in the  $\beta$ -phase region of the phase diagram. The  $\beta$ -phase is observed in the  $\beta$ -phase region of the phase diagram.

As a result, with the exception of a small number of cases, the model is able to identify the correct type of the input and output labels and the categories.

## Introduction

[illegible]

According to *Newsweek*, "a well-known Timothy Leary-type figure at Bond is an experienced helicopter chaser, a devotee of a more 'charismatic' water cult than 1960s' LSD agents like Bond championed by the company. Similar to a crop duster, 'you come in to destroy Seattle, and you reap the seeds of disaster on' half of the drug trade," *Newsweek*.



Downloaded from <http://ajphaphapublications.sagepub.com/> at 11:01 11 November 2014

<http://www.mindgarden.com>

**N**ow that the ink has dried on the map of Finland and Kazakhstan, Singapore has announced a whole range of games from its newly acquired labels, including the *U.S.A. Star Trek* that is covered elsewhere in this issue, and many others, including the *Rollercoaster*.

David DeLong starts with you lying comatose in a hospital bed while your wife comes, wangles, into a D.D. (aka, landlubber) full on your wife's nightmare, a place where you come to life. Includes various models in front of you and almost everything is lost in art you. Even conversations were very thin plastic too. C64 and PC versions on disc in June and an American one will follow.

I remember a big fight at about 1940 for growing of corn. I've been busy with working with women from 1940 to 1945, and I've been

get every wriggler and flur too can find. Each bill is worth money that can be used in the store + bank (there's a branch near you), gambled in the casino or borrowed from the bank and used in any equipment that you'll need to reach the higher parts of the tree.

Olympic Games offers four titles of fighting for the prize of gold. Two men's and two women's tournaments await your challenge in four martial arts events for the championships of Karate, Judo, Judo-Gracie, Sambo Wrestling and Budo. MMA, Aikido and FC events are due in August.

Finally, the world of the cartoons here comes, you may want to say, closer to Congress. But he saw through that little that span the Egyptian desert, temples, and even Egyptian gods. Those little cartoons to create a more "realistic" world.

owners get 40 months worth filled with traps to avoid paying the value and guards to deter. as glad you gave me a comeback.

The acquisition of these titles means that Mississippi can now draw on games from Forest Hambard Organics and Coors to support its own sense of symbolism.

Paranormal.com PC

**P**resident, already recognized as one of the major producers of quality arcade games, is set to launch some of its titles onto the PC market. *Real Captain Piz Moxy*, *The Blaster*, *Blaster: Ultra*, and *Blaster* are all to appear on PC format. OGA were born at home together and *Real Blaster* and *Captain Piz* will make an appearance in the format.

Obama takes the budget plunge.

**A**bsolutely no punishment that is less than a full day's planning and rehearsal is even budgeted here. The first of orders is not the first request will require this. Thompson's Philosophy Room is to keep the House on track here, and therefore for the first time, will give the new standard budget plan of 10, 15, and 100 to be used to help guide through the study of philosophy and other groups of events and simulations.



This is a preliminary report of the results of a study of the effects of a 12-week training program on the physical fitness of a group of 12 male subjects. The subjects were divided into two groups: a control group and an experimental group. The control group consisted of 6 subjects who did not participate in the training program, while the experimental group consisted of 6 subjects who participated in the training program. The training program consisted of a 12-week program of aerobic and anaerobic exercises. The subjects in the experimental group were trained for 12 weeks, while the subjects in the control group were not trained. The results of the study showed that the subjects in the experimental group had a significant improvement in their physical fitness compared to the subjects in the control group. The subjects in the experimental group had a significant increase in their aerobic capacity, anaerobic capacity, and maximum heart rate. The subjects in the control group had no significant change in their physical fitness. The results of this study suggest that a 12-week training program can significantly improve the physical fitness of male subjects.

**Small Business**

**S**ulfonating the latent acrylic monomer is considered to have no impact on its polymerization mechanism, as well as the control of its kinetics and its molecular weight by a better adjusted level than level of ionic intermediates and ions. Sulfonation

### Circuit Description

[illegible]

7.92 **Enthalpy of Formation**

Now, the U.S. Gold budget spin-off has launched a new 50-let budget label called Kluge, that aims to use releasing Amiga and PC games for only \$1.99. The first batch of these will include the former Christmas number one, *Outrun* (Amiga). Kluge's superb shoot-'em-up *Demolition* is *Best*

(Angebot), and the ultimate in arcade golf games, *World Leader Baseball* (Angebot, PC).

Rice is not as sensitive to cold range with 22.99 cm with varieties of *Glaxofit M*, *Manure of sh. Durova*, *Jack the Nipper*, *Minion Silver* and *Cybernet*, as well as 18.79 cm with varieties of *Glaxofit*, *Super Cycle*, *Elm*, *Red Hammer* and *World Class*.

**Journal homepage:** <http://www.elsevier.com/locate/jmb>

**J**ung, the cult movie of the '70s, is a treat to restore in a game that isn't set to be the closest for a new software house. Series 7 hopes to recreate the tropes and atmosphere in Chief Brody, Hooper (Shark expert) and Quint (shark fishermen) as well as the rest of a Great White Shark, that's increasing the inhabitants of Amity Island.



© 2005 by The McGraw-Hill Companies, Inc.

### Using Tables

**H**ere's an offer you can't refuse. Not that a, if you're a role-playing fanatic, as Oregon has at last bonded together the first three *Dungeons* to form the *Dungeon Trilogy*. They not only represent excellent value for money, but it's also the UK debut of *Dungeon II* (*Dungeon III* was the first *Dungeon* launched in the UK, followed by *IV*. Find them *AS*).

1994, 1995, 1996, 1997, 1998, 1999, 2000, 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009, 2010, 2011, 2012, 2013, 2014, 2015, 2016, 2017, 2018, 2019, 2020, 2021, 2022, 2023, 2024, 2025, 2026, 2027, 2028, 2029, 2030, 2031, 2032, 2033, 2034, 2035, 2036, 2037, 2038, 2039, 2040, 2041, 2042, 2043, 2044, 2045, 2046, 2047, 2048, 2049, 2050, 2051, 2052, 2053, 2054, 2055, 2056, 2057, 2058, 2059, 2060, 2061, 2062, 2063, 2064, 2065, 2066, 2067, 2068, 2069, 2070, 2071, 2072, 2073, 2074, 2075, 2076, 2077, 2078, 2079, 2080, 2081, 2082, 2083, 2084, 2085, 2086, 2087, 2088, 2089, 2090, 2091, 2092, 2093, 2094, 2095, 2096, 2097, 2098, 2099, 2100, 2101, 2102, 2103, 2104, 2105, 2106, 2107, 2108, 2109, 2110, 2111, 2112, 2113, 2114, 2115, 2116, 2117, 2118, 2119, 2120, 2121, 2122, 2123, 2124, 2125, 2126, 2127, 2128, 2129, 2130, 2131, 2132, 2133, 2134, 2135, 2136, 2137, 2138, 2139, 2140, 2141, 2142, 2143, 2144, 2145, 2146, 2147, 2148, 2149, 2150, 2151, 2152, 2153, 2154, 2155, 2156, 2157, 2158, 2159, 2160, 2161, 2162, 2163, 2164, 2165, 2166, 2167, 2168, 2169, 2170, 2171, 2172, 2173, 2174, 2175, 2176, 2177, 2178, 2179, 2180, 2181, 2182, 2183, 2184, 2185, 2186, 2187, 2188, 2189, 2190, 2191, 2192, 2193, 2194, 2195, 2196, 2197, 2198, 2199, 2200, 2201, 2202, 2203, 2204, 2205, 2206, 2207, 2208, 2209, 2210, 2211, 2212, 2213, 2214, 2215, 2216, 2217, 2218, 2219, 2220, 2221, 2222, 2223, 2224, 2225, 2226, 2227, 2228, 2229, 2230, 2231, 2232, 2233, 2234, 2235, 2236, 2237, 2238, 2239, 2240, 2241, 2242, 2243, 2244, 2245, 2246, 2247, 2248, 2249, 2250, 2251, 2252, 2253, 2254, 2255, 2256, 2257, 2258, 2259, 2260, 2261, 2262, 2263, 2264, 2265, 2266, 2267, 2268, 2269, 2270, 2271, 2272, 2273, 2274, 2275, 2276, 2277, 2278, 2279, 2280, 2281, 2282, 2283, 2284, 2285, 2286, 2287, 2288, 2289, 2290, 2291, 2292, 2293, 2294, 2295, 2296, 2297, 2298, 2299, 2300, 2301, 2302, 2303, 2304, 2305, 2306, 2307, 2308, 2309, 2310, 2311, 2312, 2313, 2314, 2315, 2316, 2317, 2318, 2319, 2320, 2321, 2322, 2323, 2324, 2325, 2326, 2327, 2328, 2329, 2330, 2331, 2332, 2333, 2334, 2335, 2336, 2337, 2338, 2339, 2340, 2341, 2342, 2343, 2344, 2345, 2346, 2347, 2348, 2349, 2350, 2351, 2352, 2353, 2354, 2355, 2356, 2357, 2358, 2359, 2360, 2361, 2362, 2363, 2364, 2365, 2366, 2367, 2368, 2369, 2370, 2371, 2372, 2373, 2374, 2375, 2376, 2377, 2378, 2379, 2380, 2381, 2382, 2383, 2384, 2385, 2386, 2387, 2388, 2389, 2390, 2391, 2392, 2393, 2394, 2395, 2396, 2397, 2398, 2399, 2400, 2401, 2402, 2403, 2404, 2405, 2406, 2407, 2408, 2409, 2410, 2411, 2412, 2413, 2414, 2415, 2416, 2417, 2418, 2419, 2420, 2421, 2422, 2423, 2424, 2425, 2426, 2427, 2428, 2429, 2430, 2431, 2432, 2433, 2434, 2435, 2436, 2437, 2438, 2439, 2440, 2441, 2442, 2443, 2444, 2445, 2446, 2447, 2448, 2449, 2450, 2451, 2452, 2453, 2454, 2455, 2456, 2457, 2458, 2459, 2460, 2461, 2462, 2463, 2464, 2465, 2466, 2467, 2468, 2469, 2470, 2471, 2472, 2473, 2474, 2475, 2476, 2477, 2478, 2479, 2480, 2481, 2482, 2483, 2484, 2485, 2486, 2487, 2488, 2489, 2490, 2491, 2492, 2493, 2494, 2495, 2496, 2497, 2498, 2499, 2500, 2501, 2502, 2503, 2504, 2505, 2506, 2507, 2508, 2509, 2510, 2511, 2512, 2513, 2514, 2515, 2516, 2517, 2518, 2519, 2520, 2521, 2522, 2523, 2524, 2525, 2526, 2527, 2528, 2529, 2530, 2531, 2532, 2533, 2534, 2535, 2536, 2537, 2538, 2539, 2540, 2541, 2542, 2543, 2544, 2545, 2546, 2547, 2548, 2549, 2550, 2551, 2552, 2553, 2554, 2555, 2556, 2557, 2558, 2559, 2560, 2561, 2562, 2563, 2564, 2565, 2566, 2567, 2568, 2569, 2570, 2571, 2572, 2573, 2574, 2575, 2576, 2577, 2578, 2579, 2580, 2581, 2582, 2583, 2584, 2585, 2586, 2587, 2588, 2589, 2590, 2591, 2592, 2593, 2594, 2595, 2596, 2597, 2598, 2599, 2600, 2601, 2602, 2603, 2604, 2605, 2606, 2607, 2608, 2609, 2610, 2611, 2612, 2613, 2614, 2615, 2616, 2617, 2618, 2619, 2620, 2621, 2622, 2623, 2624, 2625, 2626, 2627, 2628, 2629, 2630, 2631, 2632, 2633, 2634, 2635, 2636, 2637, 2638, 2639, 2640, 2641, 2642, 2643, 2644, 2645, 2646, 2647, 2648, 2649, 2650, 2651, 2652, 2653, 2654, 2655, 2656, 2657, 2658, 2659, 2660, 2661, 2662, 2663, 2664, 2665, 2666, 2667, 2668, 2669, 2670, 2671, 2672, 2673, 2674, 2675, 26

occurs on battle with the *Trojan* of Evi in three musical papers. In *Onesie II*—*The First Act of Darkness* you must battle with hordes of ogre-morph creatures from Mordor in the Wizard's Lair. In *Onesie*—*The Revenge of Darkness*, the land is threatened by Minus, Mordor's infamous apprentice, who has torn rifts in time in her attempt to track and sweep her father's closet. These rifts give access through which a brave adventurer may bring about her doom. Finally, in *Onesie III*—a part of *Onesie*.



**Keywords:** *Flow experience; Flow state; Flow state scale; Flow state questionnaire*

must act quickly. For Soeria is threatened by the great north Sarpaes  
 swarms from a cluster of eggs, and  
 fragments of a manuscript hint at an  
 alliance between Mordor and Mith-

## Programa Final

**A**sports with a younger brother or sister as young as an 11-year-old? You know all about Proton Paint, but he has black and white cat. Well, he and the stegosaurus are about to star on the C&A and Amiga screen courtesy of Alternative Software. For only £1.99, C&A owners can deliver the mail in Greenlands, meet people like Ted Olsen, Peter Fogg, Mike Hubbard and Dr. Gilbertus, and also attempt other tasks such as rounding up sheep (C&A owners will definitely go for this one!) - *DA*

The Amiga version will follow later (price to be announced), and will also feature Postman Pat Ludo, Snakes and Ladders and Snap. According to Alternativa the game is "mad-as-hell addictive", so is the tune. All together now, "Postman Pat Postman Pat Postman Pat and his black-and-white dog".

1000

# COMMODORE

**delivered to your door FREE!**

That's right, if you take out a year's subscription to *Your Commodore* we will send you that at no charge to your door each month at no extra charge!

Just fill in the coupon below and send it to the address given with a cheque, money order or credit card authorisation. No need to pay at the time of the subscription.

**Figure 1**

[illegible]

UK 29-30, EUROPE 229-25 MIDDLE  
 EAST 226-30 FAR EAST 226-40  
 REST OF THE WORLD 224-30  
 in USA \$27.95  
 Annual Rates on Request

Send this form with your remittance to  
IMPHENT LTD, 5 River Park Estate  
Biller Lane, BERNHARTS  
Herts. HG4 3RL, United Kingdom.

Please renew my subscription to *Your Commodore* with the same rate I enclose my check/money order for \$\_\_\_\_\_ payable to **ADAMS SPECIALIST PUBLICATIONS**.

City/Region/State/Zip number:

Model Name	Age	Sex	Species
------------	-----	-----	---------

**Keywords:** child sexual abuse; disclosure; self-blame; social support

☐ New subscriber ☐ Salesperson (new salesperson delete as applicable)

# Extending Basic

*Make your life easier by adding a trace routine to*

*Commodore Basic.*

*By Burghard-Heinry Lehmann*

**T**he initial work of writing a program is not all that hard. If you've got a fair grasp of the language you're using and know your computer quite well, you'll get something written pretty quickly. But then comes the tedious but boring and debugging!

That gets more difficult and more confusing the larger and more complex your program becomes. A computer program can easily develop into a jigsaw puzzle, and every bit has to fit exactly! A computer is a machine, and the machine has no mercy. You either get it right or you don't. If you don't, the machine will repeat the same kind of nonsense over and over a thousand times. Computer services (and often programmers, who really should know better) when confronted with a bug, open the program ten or 20 times in the vain hope that the computer will do it right eventually. But it never does!

A far more sensible approach is to get some good debugging tools. Commodore Basic has no debugging tools at all. So, in the next few articles in this series on extending the Basic of your C64, I'd like to develop a few of these. Once you know how Basic works in Basic, this becomes much less "you might think! Let's start with a trace routine."

## All About Tracing

Tracing gets activated once the program starts to execute, that is, after you've given the Run command. Before each line (or part of a line, if it is a multi-statement line) is interpreted and executed, the computer is stopped and the trace routine, which we will develop, prints the number of the line to be executed and its contents at the top of the screen.

To continue execution, just press any key - this results in the line being executed as usual. Then the computer jumps to the next line, prints that out at the top of the screen, stops again and so on. The usefulness of all this is that you can see exactly where in the program the computer is at all times. This is also called single-stepping a program, because that's exactly what the computer is doing. Normally, things happen so quickly that you barely have time to consider what exactly is going on. Tracing or single-stepping gives you the chance to take a step by step and think things through.

## Stopping the C64 in Its Tracks

We interrupt the normal program flow right at the beginning of our extended Basic routine before we look for an

extended Basic command (lines 490-510). At this point, the accumulator contains the first letter of the extended Basic command to be taken into effect. If it is an ordinary Basic command. Since we need that later on, we save it on the machine stack. Then we call the trace routine itself (lines 1300-1320).

First we test the system variable SVD, which tells us whether we're in program execution mode or direct mode. If the computer is in the program mode, SVD contains zero, otherwise it contains 128. We want to know that, because if we've just given a Basic command direct, we obviously don't need the trace facility to be activated. Therefore, we add eight away from the routine (line 1430) to move the former contents of the accumulator from the machine stack (line 1300), and compare it with

## Plotting

Next, we need to save the current print position, because if the program outputs text or graphic characters to the screen, we want them afterwards to be output in the proper print position.

By the way, I've only bothered to save the current print position and recover it later on. To make the program more functional, I advise you to save the screen colours too and print

the tracing line in a way which makes it stand out clearly. As always, I leave these lines points to you!

To save the current print position and restore our own, we use a Rom routine, called "Pint". If Pint is entered with the carry flag set, the current print position contained in the system variables SD4 and SD3 is put into X (address) and Y (down).

If Pint is entered with the carry flag clear, the value contained in X is inserted as the new print column and the value contained in Y is inserted as the new print line. This knowledge should make it easy for everybody to develop an "AT" function, something really lacking in Commodore Basic! In lines 1670-1690, we use pint to move the current print position to 231/232, and in lines 1694-1696 we restore the top line of the screen as the new current print position.

Next, we print an 80 character-long empty string to clear the top two lines of the screen (lines 1618-1630). For this we use a Rom-routine which prints any string, as long as it doesn't exceed 126 characters and is terminated with a zero. To point the computer to the string we want to print, we put the low byte of the start address of the string in the accumulator and the high byte into Y. I've decided to clear two lines, to accommodate a Basic line of any length.

Then we reset the current print position back to the beginning of the top line (lines 1670-1700). Printing the line number, which is the first thing we want to do, is also very easy. The number of the line the computer is interpreting at present is contained in the system variable LNR/1A. To print this number, we use a Rom-routine which prints any number if the accumulator contains the low byte of the number, and X its high byte.

Next, we print a separating space, using the same-to-use of all Rom-routines. With this one, you simply load the character to be printed into the accumulator and call the routine. You don't have to save any registers, because this routine does everything, including the accumulator which contains the character to be printed, before it does its work and restores everything again. (With most other Rom-routines you have to take care of this yourself!)

By the way, both the above routine and the PRINTING routine also

execute so-called "non-printable" characters, such as carriage returns, backspace, cursor movements and so on. Just put the appropriate ASCII code into the Commodore mode) into the accumulator or the string you want printed, and it does it.

## Basic Token Codes

In past articles, I have already mentioned that Commodore compresses (or "tokenizes", as the computer jargon goes) Basic keywords. One advantage of this is that it makes Basic modules more compact and thus saves memory. It also tells the computer very early when it has to deal with a Basic keyword and when it doesn't. A token code is always larger than 126, while an ordinary letter or number is smaller than 126. To put it differently, with a token code, bit 7 of the right bit of a byte (counted from 0 to 7) is set, while with an ordinary ASCII code it's clear.

Of course, this means that a Basic keyword has to be tokenized before the line is entered into the lex file, and every time the line is repeated on the screen each token has to be expanded into the keyword it stands for. When interpreting the program, the computer just uses the token code, and never expands it, because computers, unlike human beings, are far happier just to deal with numbers, rather than English words.

To expand token codes, there is a list in Rom of all Basic keywords. This list starts at location 5A/09E. Since all Basic keywords are of various lengths, the programmers of the Commodore Rom expanded each keyword from the next one by adding, once again, 126 to the last letter. Therefore, to print the last letter correctly, one has to subtract 126 from it.

Instead of this method, the Commodore which this could have used another method to build this table—they could have padded each keyword that's shorter than a certain length with spaces or zeros. This would have made it much easier to jump from entry to entry. But it would also have used up much more memory. That's why they went for the former solution.

To expand and print token codes we use a sub-routine (lines 2660-3000), because there may be several token codes in one line or line segment. This is because Basic tokens are not just commands, but also functions. It also

tolerates arithmetic operators, like "+" and "-". The reason for this is again to make them stand out clearly from other characters in the line.

## Expanding and printing a token

When we enter our sub-routine, the token code is the accumulator. First, we subtract 126 from the code to get the actual number of the keyword. Then we load into X, which will serve us as the counter (lines 2668-2688).

Next, we load the base address of the keyword table (LACB) into a zero page address so we can use subroutines later on. I use zero page 561/62, which is the first location of the floating point accumulator. It is completely safe to do this here, since we won't use the floating point accumulator (lines 2728-2736).

Now we enter the main loop, which starts by testing X. If it contains zero, we have found the keyword we were looking for. This is because we use X, which contains the number of the keyword, to count backwards. With every pass through our main loop, we decrement X by one (lines 2780-2810).

If we haven't found our keyword yet, we store Y and enter TOKEXP1 (lines 2810-2860). This loop tests each character of the next keyword to find the last character, which has 126 added to it. If it has found each character, the carry flag will be set. In the final part of the main loop (lines 2940-3010), we update the base address contained in 561/62 so that it points at the beginning of the next keyword. This is done by adding the contents of Y (the index) to it.

Once we have found the right keyword, it's printed onto the screen, again each character being tested to find the last character (lines 3050-3110). When this has been found, 126 is subtracted from its value and it too is printed (lines 3160-3166).

## Tying it all up

The rest of our trace routine is pretty simple.

Lines 2418-2440 restore the former current print position.

Finally, a Rom-routine called "Cater" is used to wait for any key being pressed (lines 3480-3490). If no key has been pressed, the zero flag is set. Otherwise zero will be clear, because CATER returns the ASCII code of the

key which has been pressed in the accumulator

As always in this series, I haven't done things as comprehensively as they could have been - I justify this by saying that I want to encourage you

to find your own solutions, but people who know me better will say that it's because I'm born idle... I haven't added a new basic command which switches text on and off. The reason you may want to introduce these com-

mands, since they allow you to trace through certain parts of a program and let other parts you are less interested in run at full speed.

Next time we'll develop some more debugging and toolkit routines.

```

10          CPG 49152
20          RMT
30
40          CHARGE  R00 R0003
50          EXECVECT R00 R0108
60          PRINT  R00 R0716
70          PRINT0  R00 R00CD
80          PRINTST R00 R081E
90          PLOT    R00 R07F0
100
110         SYMBOLTEL R00 R0000
120         ;
130         ;
140         ;
150         TURN EXTENDED BASIC ON
160         BY CHANGING VECTOR AT R0000
170
180         EXTRAS00 LDA R-PRGSTART
190             STA <EXECVECT
200             LDA R-PRGSTART
210             STA <EXECVECT
220         ;
230         RTS
240         ;
250         ;
260         ;
270         TURN EXTENDED BASIC OFF
280         BY CHANGING VECTOR AT R0108
290         BACK TO NORMAL (R07E4)
300
310         EXTRAS00 LDA R-BA7E4
320             STA <EXECVECT
330             LDA R-BA7E4
340             STA <EXECVECT
350         ;
360         RTS
370         ;
380         ;
390         ;
400         *** MAIN PROGRAM ENTRY ***
410
420         LOOK FOR EXTENDED BASIC COMMANDS
430
440         PRGSTART JNE CHARGE
450             JNE EXECSTM
460             JMP BATEE
470         ;
480         ;
490         EXECSTM  PBA
500             JNE TRACE
510             PLA
520         ;
530             CMP '0

```

```

540             BNE NEXT
550             JMP OFF RT
560         ;
570         NEXT    CMP 'C
580             BNE NORMAL
590             JNE CHARGE
600             CMP 'O
610             BNE NORMAL
620             JNE CHARGE
630             CMP 'L
640             BNE NORMAL
650
660             JNE CHARGE
670             CMP R000
680             BNE COLOR RT
690         ;
700         ; DO NORMAL ROM-ROUTINE
710
720         NORMAL   JMP BATEE
730         ;
740         ;
750         EXECUTE "COLOR" COMMAND
760
770         GET INK PARAMETER
780
790         COLOR RT JNE CHARGE
800             JNE BADA8A
810             JNE B00FF
820
830         CHANGE INK COLOUR
840
850             STY 04E
860         ;
870         GET PAPER PARAMETER
880
890             JNE CHARGE
900             JNE BADA8A
910             JNE B00FF
920
930         CHANGE PAPER COLOUR
940
950             STY 01261
960         ;
970         GET BORDER PARAMETER
980
990             JNE CHARGE
1000            JNE BADA8A
1010            JNE B00FF
1020
1030         CHANGE BORDER COLOUR
1040         ;

```

# PROGRAMMING

<pre> 1000      STY 2000 1001 1002      JUMP TO REST OF NON-ROUTINE 1003 1004      RTE 1005 1006 1007 1008 1009      TEST FOR REST OF "OFF" 1010 1011 1012 1013      OFF RT      JSE CHANGET 1014                  CMP 'F' 1015                  BEQ OFF RT1 1016                  JMP NORMAL1 1017      OFF RT1     JSE CHANGET 1018                  CMP 'F' 1019                  BEQ OFF RT2 1020                  JMP NORMAL1 1021 1022 1023      EXECUTE "OFF" COMMAND 1024 1025 1026      OFF RT2     JSE EXTRADOFF 1027 1028      GET NEXT CHARACTER AND 1029      JUMP TO REST OF NON-ROUTINE 1030 1031 1032      JSE CHANGET 1033      RTE 1034 1035 1036 1037      TRACE ROUTINE 1038 1039      IF DIRECT MODE, EXIT AT ONCE 1040 1041 1042      TRACE      LDA #PD 1043                  CMP #RND 1044                  BNE TRACE1 1045                  RTE 1046 1047      SAVE CURRENT PRINT POSITION 1048 1049      TRACE1     SEC 1050                  JSE PLOT 1051                  STX 201 1052                  STY 202 1053 1054      PLOT TOP LINE PRINT POSITION 1055 1056 1057      CLC 1058      LDX #0 1059      LDY #0 1060      JSE PLOT 1061 1062 1063      CLEAR TOP TWO LINES 1064 1065 1066      LDA #EMPTYLINE 1067      LDY #EMPTYLINE 1068      JSE PRINTSTR 1069 </pre>	<pre> 1650      PLOT TOP LINE PRINT POSITION 1651 1652 1653      CLC 1654      LDX #0 1655      LDY #0 1656      JSE PLOT 1657 1658 1659      PRINT LINE NUMBER 1660 1661 1662      LDA 20A 1663      LSR 210 1664      JSE PRINTNO 1665 1666 1667      PRINT ONE SPACE 1668 1669 1670      LDA 202 1671      JSE PRINT 1672 1673 1674      GET ADDRESS OF BASIC TOKEN AND 1675      PRINT IT 1676 1677 1678      LDY #0 1679      LDA (27A),Y 1680 1681      JSE TOKENISE 1682 1683 1684      PUT CHANGET ADDRESS INTO 203/204 1685      AND INCREMENT BY ONE 1686 1687 1688      LDA (27A) 1689      STA (203) 1690      LDA (27A) 1691      STA (204) 1692 1693 1694      INC (203) 1695      INC (204) 1696      INC (205) 1697 1698      PRINT REST OF LINE 1699 1700 1701      TRACE      LDY #0 1702      PRINTLOOP  LDA (201),Y 1703                  BEQ LINEEND 1704                  CMP ' ' 1705                  BEQ LINEEND 1706                  CMP #128 1707                  BCC PRINTLOOP 1708 1709 1710      IF TOKEN, SAVE Y AND PRINT TOKEN 1711 1712 1713      DNY 1714      STY 203 1715 1716 1717      JSE TOKENER 1718 1719 1720      POINT AT CHARACTER AFTER TOKEN 1721      AND LOOP BACK 1722 1723 1724      CLC </pre>
--	---



```

2240 LDA #0
2250 ADC #250
2260 STA #250
2270 BCC TRACE2
2280 INC #250
2290 BNE TRACE2
2300
2310 PRINT ORDINARY CHARACTER, INCR.
2320 INDEX AND LOOP BACK
2330
2340 PRINTLOOP JSR PRINT
2350 INY
2360 BNE PRINTLOOP
2370
2380 END OF LINE RE-PLOT OLD PRINT
2390 POSITION
2400

```

```

2410 LINEEND CLC
2420 LDA #51
2430 LDY #32
2440 JSR PLOT
2450
2460 WAIT FOR KEYPRESS.
2470
2480 WAIT JSR $FEE4
2490 BEQ WAIT
2500
2510 IF KEY PRESSED, WAIT

```

```

2520
2530 RTS
2540
2550
2560
2570 NORMAL JMP $A7E2
2580
2590

```

```

2600
2610 TOKEN SUBROUTINE
2620
2630 CALCULATE TOKEN NUMBER AND
2640 STORE IT IN X
2650
2660 TOKENNR SEC
2670 SEC #120
2680 TAC
2690
2700 INITIATE REG. OF TOKEN TABLE
2710
2720 LDA #PAGE
2730 STA #0

```

```

2740 LDA #PAGE
2750 STA #0
2760
2770 IF X=0, TOKEN FOUND.
2780
2790 TOKENLOOP CPA #0
2800 BEQ PRINTOKEN
2810 BEQ
2820
2830 FIND END OF TOKEN
2840
2850 LDY #0
2860 TOKENLP1 LDA (#0),Y
2870 CMP #120
2880 BCS ENDTOKEN
2890 INY
2900 BNE TOKENLP1
2910
2920 POINT TO REG. OF NEXT TOKEN
2930
2940 ENDTOKEN CLC
2950 INY
2960 TRA
2970 ADC #0
2980 STA #0
2990 BCC TOKENLOOP
3000 INC #0
3010 JMP TOKENLOOP
3020

```

```

3030 PRINT BASIC WORD
3040
3050 PRINTOKEN LDY #0
3060 PRINTOK1 LDA (#0),Y
3070 CMP #120
3080 BCS TOKENEND
3090 JSR PRINT
3100 INY
3110 BNE PRINTOK1
3120
3130 PRINT LAST CHAR. OF BASIC WORD,
3140 AFTER HAVING SUBTRACTED 120
3150
3160 TOKENEND SEC
3170 SEC #120
3180 JSR PRINT
3190
3200 RTS
3210
3220
3230
3240 EMPTYLINE RST
3250 RST
3260 RST
3270 RST
3280 RST

```





to the Astoroute maps. The examples supplied are pubs and hotels, but you could add anything from distributors to branches and software houses to customers.

For each entry, you can compile a small text entry that appears when the location is selected by mouse. For example clicking on the software house in Sams would reveal something like: **New Base**, **insiders of Astoroute and Astoroute Plus** and its expansion modules. Tel: 0784 468077. Fax: 0784 468282.

That's not all - the **Questor** also allows you to add detailed departure and arrival instructions that are incorporated into the table so that people will always know how to find you.

The **peripos** module includes the location of the center of each of the

1700 post code regions, so that you can pinpoint your customers accurately and efficiently. This efficiency can be enhanced by adding the optimization and costing module that will automatically plot the most efficient route between calls and calculate costs based on linearly and fuel-dependent rates.

There are modules on the way to customize Astoroute Plus to your specific needs - these will include a symbol editor to add symbols to the maps (for use with the **Questor** hardware), that plots destinations (the same travelling time away from your start position (L.O. what's 45 minutes from Birmingham - what for distributors), a restriction module that plans routes to avoid low bridges, width and weight limits, and an overlay of periodic areas and county boundaries.

**Astoroute** and **Astoroute Plus** tell have many users, ranging from computer users wanting to improve family and friends, to travelling salesmen, date organising events and small and not-so-small businesses. It will run on most PCs, but becomes really useful when loaded into a portable that can be used while you're in the car.

The basic **Astoroute** costs £120, which is quite reasonable considering the mass of data it stores and processes. **Astoroute Plus** is aimed more at the professional user and costs a £299 + V.A.T. price tag, with the modules costing between £149 and £199.

#### Findings

**Title:** *Astoroute Supplier*, New Base, Unit 18, Central Trading Estate, Sams, Middlesex. TW7B 4EE. Tel: 0784 468077.

# PC Corner

*Commodore has joined just about every other computer manufacturer and started producing PC Clones. We start a regular column for Commodore PC owners.*

If you own a Commodore 64, you may be considering changing your machine for an Atari, or another 16 bit machine. However, you may still be confused as to what's available, or the Atari may not be the right machine for you. What are the alternatives? The Atari ST is one, but you could also consider getting a PC compatible. This might seem a strange choice, but settling for what has become the de facto industry standard does in fact make a lot of sense. An explanation of why you should take such a course is necessary, so I will attempt to make clear both the pros and cons in this article.

## The IBM Standard

The term *IBM PC compatible* refers to a hardware standard, that is a machine which has certain minimum specifications. These are an 8088 or 8086 CPU running at a speed of 4.7Mhz, a minimum of 64k RAM, a video card capable of displaying at least 64x25 rows of text, and one double-sided 40 track disk drive with 10Mk capacity. To be fully compatible with the IBM standard, the ROM chips inside the machine should also be the same as IBM's.

You may wonder how manufacturers have survived the threat of legal action from one of the world's biggest companies. The answer is, only at their indulgence, by not making copies directly and by using a technique called "reverse engineering." Even basic items such as printer ports are not part of the spec. However, one of the great advantages of the standard is that a large number of expansion slots come in the basic machine, so you can buy almost anything, and just plug it in. Don't worry if you're not quite sure

what all this means—everything should become clear as you read on.

## MSDOS

Also called PCdos or true IBM machines, MSDos is the heart of a PC compatible. Without it, you cannot do anything at all.

MSDos is usually loaded from disk, but some portables, for example, have a version in ROM. MSDos is responsible for handling all input and output. This means the screen, printer, keyboard or disk drive. The software that does all this is usually loaded once, at switch on and then resides in the machine until it is switched off. The other part of MSDos is made up of a number of *Transient* commands. Of these, by far the most important is *Command.com*, the so-called command processor. This program is like the Basic interpreter, in that it sets its memory intercepting your keypresses, and turning them into something the computer can understand.

All MSDos commands are given from a prompt which initially consists of just the disk drive letter, and a > symbol, eg.

A > or C >

This can be changed to suit your needs to show the date, for example. Here you will do most work, launching programs, formatting disks, and keeping track of data. Typing the built-in command *DIR* at this prompt will reveal what's on disk. *Pt /* shows the directory listings of the MSDos system disk. You will see that files on the disk all have up to eight letters, then a dot(.) followed by an extension of up to three letters. How a file is treated by MSDos depends on these letters. *EXE* or *COM* after the dot means a program that will run from

the A > prompt. *BAT* tells MSDos that the file is a text file which can be treated as a list of commands. One of these *BAT* files, *Autoexec.bat* is special, because it executes automatically on startup. This means you can set your machine up to suit you, in the knowledge that every time you switch it on, it will do the same.

One other important thing to understand about MSDos is that it is a hierarchical filing system, and so within one directory you can have not only files, but more directories, and so on. Organising files this way is essential if you have a hard disk, of which more later.

## Programming the PC

The surprising thing about PC compatibility is they almost never have Basic built into them. Instead Basic comes as just another program that can be run. *GWBasic* is generally accepted as the standard here and comes with most machines. An interpreted language, it is generally considered to be well featured and easy for beginners to get to grips with, but it is slow. However, numerous languages are available, and if you wish to program professionally you should consider one of the many compilers available. Assembly language programming is also well covered for, both commercially, and in the public domain. PCs do of course have the *BATCH* language built into them. This is very simple, but experienced batch language programmers can do some surprising things with it.

## Software

Upwards of one million packages have been written for the machine, and this number is growing all the time. All the big software houses write for the



PC, and some packages such as Lotus 123, dBase III, and Wordperfect are standards in their own right. Standards are high, but be warned, unless you have an expensive machine capable of running Windows, most software is still text-based. The quality of games software can be variable, with only the more recent stuff for EGA graphics adapters being really exciting. This will improve as more software houses take to the PC. Public domain software is available in large quantities, so even the impressionists will find something to meet their needs.

## Graphics

Probably the most perplexing aspect of PCs is that of graphics and display adapters. Unfortunately there is no single standard, and of those which do exist there are many variations. Also, many programs such as Adobe Page-maker require a minimum configuration before they will run. In Page-maker's case an EGA adapter and colour monitor are required before it will run.

The problem has arisen because, in the beginning the display standard was 640x350 text in monochrome. Only later did provision for colour and

graphics arise. All PCs come with some kind of graphics adaptor, usually on a plugin card and a monitor, so it is important to get the right display for your needs. The following list should give an idea of what's available.

The list shows just how desirable some combinations can be.

The drawback, however is cost. With a suitable VGA setup costing as much or more than most low cost PCs, many users have to settle for something less. If you can afford it, I would suggest an EGA is the best overall value for money.

## What to look for when buying a PC

Thankfully, modern PC compatibles far exceed the specifications of their ancestors. You should expect memory of 512 or 640K.

A turbo processor running at 8 or

10MHz is advisable. For those with ample cash, an AT class machine which uses the fast 80386 processor could be acquired. Serial and parallel ports should be built in and at least two or three expansion slots should be available.

An Personal Computers are dual board, a minimum of two 386 3 1/4 inch, or one 720 3.5 inch drive should be included. Hard disks are cheap in the PC world, and are really worth the money at about 200 pounds for 28Mbytes.

Check also what software is bundled with the machine. An integrated package such as *Atkey*, or *Works* could be all you need for the first six months or so.

## Conclusion

PC compatibles are a safe option. You won't ever set the world alight with one, but at the same time nobody is going to laugh at you for getting one. A very wide range of price and performance is covered, so the chances are there will definitely be one to match your budget.

Commodore have a range typical of many manufacturers, ranging from budget 8088 machines, to fast 80386's with huge amounts of memory and disk storage.

## Get in Touch

PC Corner is designed to be a forum for all users of the Commodore PC range, but it's important to remember that without your input, it simply won't work. We want PC Corner to work, but we need your help, so if you have any comments, tips or general PC queries, please get in touch with us at

### PC Corner

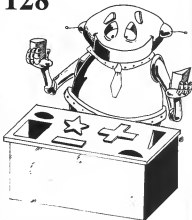
Four Commodore  
Argus House  
Boundary Way  
Hemel Hempstead  
Herts  
HP2 3ST

Adaptor	Resolution	Use
MDA	640x350(mon)	Wordprocessing etc
HGA	as above plus 720x400 col	as above plus CAD etc
CGA	320x200*4 & 640x300*2 col	general & games
EGA	up to 640x350*16 col	Hi res graphics/games
VGA	up to 640x480*64 col	DTP etc

# Reasoning on the 128

*The first part of a series that may help you turn your humble 128 computer into an expert system*

*By Paul Schofield*



I find it surprising that there is still very little AI software available for Commodore Machines. One likely reason for this phenomenon is that many people have been given the impression that AI applications can only be programmed in either PROLOG or LISP. While these languages are certainly very good tools for this type of application, it's quite feasible to employ almost any programming language. A second problem is that AI applications tend to be very data intensive, and the combination of limited memory and slow disks is somewhat restrictive.

Despite this, examples of random AI techniques are used in many C64 adventure games, some of which feature quite sophisticated natural language processors. In this article we will look at another application area, the *Commodore Expert System*. All the programs are written entirely in Commodore Basic 7.0, but allow quite a wide range of simple Expert Systems to be created. Before looking at the first program, however, it's quite natural to look at the features of traditional AI tools and the typical applications of expert systems.

## PROLOG

PROLOG is quite unlike Basic and other popular computing languages. Whereas traditional programming languages are concerned with arithmetic calculations, PROLOG is

designed to tackle problems by logical deduction. The language contains two basic types of statements. The first type is used to establish relationships between data items, which are then used to create a so-called Knowledge Base. The second type of statement is then used to interrogate the Knowledge Base to determine answers to particular problems.

The great power of PROLOG is that it doesn't need to understand the relationships defined. For example, if a statement wants to organize its items to avoid dishes that include one of season fruits and vegetables, they can

use statements like *expensive (raisinberries, January)*.

This condition can then be used by

*T = expensive (strawberries, January).*

to which PROLOG will answer YES. Not very exciting, but if strawberries is replaced by a variable, then it's possible to look at all things which are expensive in January.

## LISP

LISP stands for LInk Processing, and can perform similar types of operations,

but in this case all the conditions must be presented in the form of a list. The language provides operations for moving through the list structure, and taking different paths according to the value of individual elements in the list. For a typical AI application a LISP list looks rather bewildering, with many layers of nested brackets used to define the hierarchy of sublists. However, this approach is conceptually simpler, as each list element is at the lowest level in an English language statement.

There are the questions the user is asked and the user's responses (TRUE or FALSE) determine the next list element presented. CPUG members, who are interested in LISP should request disk CL15 from the C128 library, which contains a C/PM Mode Lisp Interpreter. Be warned though, you won't get very far without a text book. Alternatively, you can find quite similar facilities within the C64 Logo, together with a very incomplete example of Knowledge Base on the Valley Road.

## What's an Expert System?

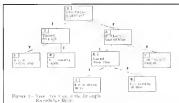
While the two favourite languages of the AI community can provide many useful clues, it's also useful to look at the basic concept of an Expert System. As the name suggests, the best way to start is to find someone with a lot of experience of tackling a complex problem. You then spend a good deal of time asking them to run through how they approach the problem step by step.

This will highlight a sequence of questions they ask to test their problem, and how they proceed according to the results obtained. It's a well-known fact that all AI Engineers have very terrible ears, and although mine has proved very reliable, this is still quite a good place to start.

There are a very large number of problems, which can be seen to mean that you're not getting optimum performance from your car. For most of us, the most serious problem is when the car won't start, and so this should be the first consideration of our computerised car maintenance expert. The easiest way is if it asks:

### CAR WON'T START?

An affirmative answer selects



analysis of this problem, otherwise other problems are considered. If we consider just the case of the car not starting, we now investigate the simplest reason first, so as to minimise the amount of work involved in isolating the cause of the problem. The next question is likely to be:

### FUEL TANK IS EMPTY?

and if this is not the case,

### BATTERY IS FLAT?

A negative response at this point will lead to a series of questions to trace through the fuel and electrical systems. Car maintenance is a good example of a domain-specific expert system, because it clearly demonstrates the two big advantages of an expert system:

1. A guide through the steps to analyse a problem.
2. Ultimately identifies a solution.

## Fuzzy Logic

The part of this example we've looked at so far works very nicely, as it's quite simple to answer TRUE or FALSE to each question. If we proceed a little further, we will encounter the question:

### IS SPARK GOOD ON ALL PLUGS?

We can easily test whether there is a spark, but how do we know if it's adequate? Much research is currently being done into fuzzy logic. As there's no measurable divide between a good and inadequate spark, a probability is associated with the measured value within the critical range, and can be used in conjunction with probabilities for other components to give an assessment of whether the combination works.

At this stage we now have a non-deterministic expert system, which can't give a definite answer, just the most probable answer. As most home motion mechanisms would have no means of accurately measuring the strength of the spark, these motions can be ignored, and the somewhat approximate question employed. This fits in rather nicely with the LISP list approach.

## REASON 128

REASON 128 is an interactive Expert System tool which enables you to create, modify, store and interrogate a moderately sized knowledge base. It reflects the fact that almost all knowledge bases can be represented as a simple tree structure.

This representation greatly simplifies the problem of constructing a Knowledge Base, as each statement need only be associated with two pointers to the next statement to be displayed, according to whether or not the previous one was TRUE or FALSE. This is of course a double in Pascal or C, as it's just a matter of defining records to create a binary tree. It can also be achieved very simply in Basic using three arrays.

## Designing a Knowledge Base

The design of a Knowledge Base requires two things:

1. some knowledge of the subject (e.g. text book)
2. a large sheet of paper.

The reference text provides the information on the questions/answers and eventually the conclusions and the paper is used to arrange them in a tree structure.

Let's move on now out of the garage and into the chemistry lab. At the back of a shelf we find an unlabeled bottle containing a clear liquid. This could be water, sulphuric acid, caustic soda or numerous other things, and it would be nice to know which one it is. To keep this example reasonably short, we'll only consider tests related to the acidity or alkalinity of the liquid.

We can do very simple tests using litmus to determine whether it's acidic or alkaline, and also make a subjective analysis of these tests. This can be defined in terms of four questions and five conclusions, with those arranged into a simple tree structure as shown in Figure 1. If we get a positive result to test 1, we can ignore the right-hand half of the tree and proceed to question 2. The answer to this results in conclusion 3 or 4.

Working with REASON  
126

Once you have a tree diagram like the one in Figure 8, you're ready to use REASONER [28]. Run programs one, and a list of six options is displayed. Select 1 to create a new Knowledge Base. This will then prompt for the first question to be asked. Type in the last string you wish to be displayed, and press RETURN. It then prompts for the next questions to be asked, according to whether the answer was TRUE or FALSE. When you reach the end of a branch, these are coded as 0 to indicate that this is a conclusion rather than a choice.

When you've entered all statements on your form, turn to the next

時間 (時刻)	日 次	観測 項目 (内容)	観測 結果 (内容)
1 1 0	14.0.1	1. 観測 項目 (内容)	1. 観測 結果 (内容)
1 1 0	14.0.1	2. 観測 項目 (内容)	2. 観測 結果 (内容)
1 1 0	14.0.1	3. 観測 項目 (内容)	3. 観測 結果 (内容)
1 1 0	14.0.1	4. 観測 項目 (内容)	4. 観測 結果 (内容)
1 1 0	14.0.1	5. 観測 項目 (内容)	5. 観測 結果 (内容)
1 1 0	14.0.1	6. 観測 項目 (内容)	6. 観測 結果 (内容)
1 1 0	14.0.1	7. 観測 項目 (内容)	7. 観測 結果 (内容)
1 1 0	14.0.1	8. 観測 項目 (内容)	8. 観測 結果 (内容)
1 1 0	14.0.1	9. 観測 項目 (内容)	9. 観測 結果 (内容)
1 1 0	14.0.1	10. 観測 項目 (内容)	10. 観測 結果 (内容)

Figure 1. Characteristic Frequency-Related Noise

prompt and you will then be asked if you want a hard copy of the Knowledge Base. This is in the format shown in Figure 2, and is useful both as a check, and for future reference if you wish to make any changes. You're then returned to the main menu:

All this points to is it is advisable to use option 3 to save the Knowledge Base on disk. If you don't have a formatted disk then, use option 3 first, when a Knowledge Base has been saved, it can be reloaded using option 1. It's important to note that the Base routine adds the extension ".KBF", and this part of the filename should not be specified when using Load.

Finally, option 4 is used to incorporate your Knowledge Base. Just keep answering T(rue) or F(alse) to the questions until REASON highlights an inconsistency.

### Probleme und Limitationen

For reasonably small applications, you'll find REASON 128 both efficient and easy to use. For more complex applications, however, it is often difficult to construct a complete tree at the outset. It can also mean that

you have to use an identical quantity at several parts of the tree, that resulting in a whole tree.

For such applications, it would be much more to have a system that allowed you to define just the facts which you know about possible solutions. Later in the article, we'll look at three programming languages that make such a system. You'll still find REASON useful, as such systems frequently produce multiple solutions. REASON can help in developing the extensions to eliminate these and improving the efficiency of the data structures.

### Playing Games

Although REASON was designed primarily for setting up and interpreting simple tree structures, it need not be limited to such applications. One quite interesting area is repetitive applications that can occur in many sample games. Figure 3 is a listing of the REASON knowledge base for a Naughts and Crosses system. Like many such games, the strategy is complicated only for the opening moves, after which it is simply a verification of a very small set of rules.

```

404000 128 - QUESTIONS * CONCLUSIONS
C 1 1 PLACING X'S BY GOOD FIRST: T-1 2 3 F-1 4 5
C 2 1 PUT X IN ANY CORNER - NEXT MOVE T-1 2 3 F-1 4 5
C 2 2 B PUT IN SQUARE AT CORNER OF CONTRA T-1 5 3 F-1 6 5
C 3 1 IF YOU DON'T WANT MY WLP THEN GIVE ME T-1 6 3 F-1 1 4
C 4 1 PUT X IN OPPOSITE CORNER TO FIRST: X - NEXT MOVE T-1 6 3 F-1 4 5
C 4 2 PUT X IN AN ADJACENT CORNER FORMING O'S TO BLACK - HE' HAVE T-1 7 3 F-
C 4 3
C 5 1 O'S BLOCKED LINE T-1 6 3 F-1 2 3
C 6 1 O'S FIRST MOVE WAS TO CORNER OPPOSITE FIRST: X T-1 6 3 F-1 4 5
C 6 2 PLACE X TO COMPLETE LINE - YOU NOW THANKS TO B DON'T ADVISE T-1 2 3 F-
C 6 3
C 6 4 PUT X IN LAST EMPTY CORNER - NEXT MOVE T-1 5 3 F-1 4 5
C 6 5 CAN COMPLETE LINE OF O'S T-1 5 3 F-1 4 5
C 7 1 O'S CAN COMPLETE LINE T-1 5 3 F-1 4 5
C 8 1 CENTRE SQUARE IS VACANT - O'S 6 3 F-1 4 5
C 8 2 A CORNER IS VACANT T-1 5 3 F-1 4 5
C 8 3 BLACK C TO BLOCK LINE OF O'S - NEXT MOVE T-1 6 3 F-1 4 5
C 8 4 PUT X IN CENTRE - NEXT MOVE T-1 1 3 F-1 4 5

```



```

17 1 PUT Y IN VACANT CORNER - NEXT MOVE THIS IS 2 F=10 4
18 2 PUT Y IN ANY EMPTY SQUARE - NEXT MOVE T=10 10 2 F=10 4
19 3 2 OR MORE SQUARES LEFT - T=1 11 2 F=10 20 1
20 3 SAME CORNER - 1 ONLY PROMISED NOT TO LOSE - 1 12 2 F=10 3
21 3 FIRST X WAS PUT IN CORNER THIS IS 3 13 20 1
22 3 PUT 0 IN ANY CORNER - NEXT MOVE T=10 20 2 F=10 4
23 3 PUT 0 IN CENTRE - NEXT MOVE T=10 20 3 F=10 4
24 3 CAN COMPLETE A LINE OF 0 - T=1 20 2 F=10 20 2
25 3 PLACE 0 TO COMPLETE LINE - YOU WIN THANKS TO 1 20 2 F=10 2 F
END 0 1
26 1 1'S CAN COMPLETE LINE T=1 20 2 F=10 20 2
27 3 PLACE 0 TO COMPLETE LINE - NEXT MOVE T=10 20 3 F=10 4
28 3 4 CORNER IS VACANT T=1 20 2 F=10 20 2
29 3 PUT 0 IN CORNER CLOSEST TO FIRST X'S - NEXT MOVE T=1 21 2 F=10 4
30 3 PUT 0 IN ANY EMPTY SQUARE - NEXT MOVE T=10 21 2 F=10 4
31 3 2 OR MORE SQUARES LEFT - 1 21 2 F=10 20 2
32 3 PUT 0 IN CENTRE - NEXT MOVE T=1 21 2 F=10 4
33 3 TWO 1'S IN OPPOSITE CORNERS T=1 21 2 F=10 20 2
34 3 PUT 0 NEXT TO ONE OF THE 1'S - NEXT MOVE T=1 21 2 F=10 4

```

Figure 3: Minigolf and Crosses Knowledge Base

## REASONING ON THE 128



### PROGRAM ONE

```

IF MORE THAN
ONE NEW QUESTION DISK ACCESS THEN
OUTLINE
1000 PRINT "CURRENT CORN IN - 0
NEXT SLOW - STAYING FAST
1010 PRINT "CURRENT DATA SIZE AND
PROMISE (OFFER) TO CONTINUE FBI
00
1020 GOTO 10 IF YES THEN 10
1030 PRINT "DATA CORN, LINE
COUNT - PRINT ISLAND CRITIC FAST
PRINT INITIATION OF KNOWLEDGE &
ADD 100 CHARACTERS TO
1040 IF LEFT(10) THEN PRINT LEFT
105 101
1050 PRINT "END
1060 RETURN
1070 NEW KNOWLEDGE BASE
1080 PRINT "LOAD KNOWLEDGE NEW
KNOWLEDGE BASE (OFF) - PRINT "PRINT 1
INITIALIZING VARIABLES - PLEASE W
AIT 5 SECONDS" PRINT
1090 FOR N=1 TO 50
1100 UNTIL (PRINT) PRINT N
1110 NEXT N
1120 PRINT "END 0000 0000 0000 0
00000000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000
1130 PRINT "0 1 - THE PRINT ON
THE FOR - 10 1000 0 1 1000
0000 0000 0000 0000 0000 0000
1140 PRINT "END 1000 0000 0000 0000
1150 PRINT "IF THIS WERE 10 10
IF MORE - 1000 0000 0000 0000
1160 10 1000 0000 0000 0000 0000

```

```

1170 1 PRINT "CURRENT CORN IN 0
1180 PRINT "IF THIS WERE 10 10
1190 IF MORE - 1000 0000 0000 0000
1200 10 1000 0000 0000 0000 0000
1210 PRINT "END 1000 0000 0000 0000
1220 10 1000 0000 0000 0000 0000
1230 PRINT "END 1000 0000 0000 0000
1240 10 1000 0000 0000 0000 0000
1250 PRINT "END 1000 0000 0000 0000
1260 10 1000 0000 0000 0000 0000
1270 PRINT "END 1000 0000 0000 0000
1280 10 1000 0000 0000 0000 0000
1290 PRINT "END 1000 0000 0000 0000
1300 10 1000 0000 0000 0000 0000
1310 PRINT "END 1000 0000 0000 0000
1320 10 1000 0000 0000 0000 0000
1330 PRINT "END 1000 0000 0000 0000
1340 10 1000 0000 0000 0000 0000
1350 PRINT "END 1000 0000 0000 0000
1360 10 1000 0000 0000 0000 0000
1370 PRINT "END 1000 0000 0000 0000
1380 10 1000 0000 0000 0000 0000
1390 PRINT "END 1000 0000 0000 0000
1400 10 1000 0000 0000 0000 0000
1410 PRINT "END 1000 0000 0000 0000
1420 10 1000 0000 0000 0000 0000
1430 PRINT "END 1000 0000 0000 0000
1440 10 1000 0000 0000 0000 0000
1450 PRINT "END 1000 0000 0000 0000
1460 10 1000 0000 0000 0000 0000
1470 PRINT "END 1000 0000 0000 0000
1480 10 1000 0000 0000 0000 0000
1490 PRINT "END 1000 0000 0000 0000
1500 10 1000 0000 0000 0000 0000

```

```

1510 PRINT "END 1000 0000 0000 0000
1520 10 1000 0000 0000 0000 0000
1530 PRINT "END 1000 0000 0000 0000
1540 10 1000 0000 0000 0000 0000
1550 PRINT "END 1000 0000 0000 0000
1560 10 1000 0000 0000 0000 0000
1570 PRINT "END 1000 0000 0000 0000
1580 10 1000 0000 0000 0000 0000
1590 PRINT "END 1000 0000 0000 0000
1600 10 1000 0000 0000 0000 0000
1610 PRINT "END 1000 0000 0000 0000
1620 10 1000 0000 0000 0000 0000
1630 PRINT "END 1000 0000 0000 0000
1640 10 1000 0000 0000 0000 0000
1650 PRINT "END 1000 0000 0000 0000
1660 10 1000 0000 0000 0000 0000
1670 PRINT "END 1000 0000 0000 0000
1680 10 1000 0000 0000 0000 0000
1690 PRINT "END 1000 0000 0000 0000
1700 10 1000 0000 0000 0000 0000
1710 PRINT "END 1000 0000 0000 0000
1720 10 1000 0000 0000 0000 0000
1730 PRINT "END 1000 0000 0000 0000
1740 10 1000 0000 0000 0000 0000
1750 PRINT "END 1000 0000 0000 0000
1760 10 1000 0000 0000 0000 0000
1770 PRINT "END 1000 0000 0000 0000
1780 10 1000 0000 0000 0000 0000
1790 PRINT "END 1000 0000 0000 0000
1800 10 1000 0000 0000 0000 0000

```





# Thesaurus

*Have you ever been stuck for words while writing a letter or document? Then perhaps THESAURUS can help you*

*By Norman Hart*

**A**s you may have guessed, this program's function is to present groups of words with similar meanings, derived from an alphabetically generated list. All procedures are menu driven, so even if you don't possess great typing skills, mistakes are impossible.

The database is sorted out in such a way that searches are unnecessary; the results of your requests are derived directly from the array in memory, and are displayed instantly on the screen — time after time. The program is of course crash-proof, with [RUN/STOP-RESTORE] your only means of escape.

## How it works

The program begins by POKing an alphabetical sort into location 49152 (the value of AS in line 183 — this can be changed if you wish), the data for this is held in lines 300 to 330. By this way, do take care with the values, since the sort is a very essential procedure.

The main database (lines 1846 onwards) is then read up and dispatched (lines 360 A:370) before being committed to memory as the AS() array. This first array is a simple representation of the database, word for word, and acts as the resource for your requests. In order for this pool of information to be tapped, a controlling array is required — namely the LH() array.

This second array is created on the screen, for example "ABATE 120" is AS(120), together with its own subscript, is input into memory as part of the LH() array — see INPUT 1, LH() in line 468. Thus we eventually end up with two arrays — the AS() representing the database, and LH(),

representing the AS(), together with the appropriate AS() subscripts. MINUS the subscripts. Check line 360 for INPUT 1 to see how I created out the subscripts in the DIM LH() (DHE-1) as line 364, and check line 460 to see how I created their appearance on the screen as line 450.

Once the LH() array is completed,

the STS AS in line 438 sorts the array into alphabetical order. Despite its machine code pedigree, the sort has a lot of work to do, as he prepared for a wait of up to three minutes in this large program example! With that important job done, the next task is to create a special Control String that allows the user to call up the word list at whatever point he or she wishes on the screen.

This string (CS) is a compilation of array position numbers for each fresh alphabetical change that occurs in the LH() array. In other words, if the user presses the letter "D", the list on the screen will begin with words starting with the letter "D". This listing works for the entire alphabetical range. The SPACE bar also allows the list to be paged forward and back in follow-on order. In this way words can be increased very quickly. Study lines 500 to 560 to see how the Control String (CS) is created.



Now that the program has initiated itself, an alphabetical list appears on the screen - this always begins with words starting with the letter "A". This list appears alongside all the necessary prompts and safe guards. All you have to do is press the appropriate letter key and, if needed, the SPACE bar. COME/SPACE if you wish to page backwards, then press the RETURN key and the top left-hand word will appear in Row 1.

Now turn the up and down CURSOR keys to position the Reverse highlight over the word you wish to investigate. Press the RETURN key again, and your list of associated words will appear on the screen. For example, the word **ROD** will reveal words as GIANT, MASSIVE, ENORMOUS, etc.. The F1 key allows you to return to the same previous position in the word list.

## Primer Quiz

© 2004 Blackwell Publishing Ltd *Journal of Internal Medicine* 255: 115–122

the society, but the premier has its share of responsibilities, and there are three types of trust out at your disposal.

1. The full Data Base as it appears at the end of the program. Tap the F3 key, then immediately hold down the CTRL key until the printer starts.
2. The Alphabetical List (together with the appropriate A/E) subscripts. This print out begins from the alphabetical appearance on the screen. Tap the F4 key, then immediately hold down the CTRL key until the printer starts.
3. A print out of your selected group of words. From the F3 key only.

Print items 1 & 2 are only available while the Alphabetical List is on the screen. They are protected from casual use by employing the CTRL key as described, and also there are a lack of graphics on the screen - I have noticed that some only to you, the programmer. (What may I do, and

accessible while your selected group of words appears on the screen. All printed out with a word count, and printing can be abandoned by holding down any key on the C-64. Of course, if you don't have a printer, except for the C-64 and lines 780, 790 & 1410.

As for the database itself it need not be restricted to a Timeseries. Any groups of words, ideas or items that share a common interest can be enclosed — parts for equipment perhaps, or even foreign language translations. Just remember to enclose them by asterisks as I have done in the enclosed example and flush the data statements with a double asterisk.

Not only will you find this program useful for your own literary efforts, but any younger members of the family will certainly benefit too. My eight-year-old son has told me to tell you!

# Listings

© 2005 Blackwell Publishing Ltd, *Journal of Internal Medicine* 258: 103–110

[illegible][illegible][illegible]

[illegible]

YOUR DOWNLOAD request: 780  
780

Age Group	Percentage
18-24	10%
25-34	25%
35-44	35%
45-54	20%
55-64	10%
65-74	5%
75-84	5%
85+	5%

NOTES: CONFIDENTIAL, INTERNAL USE

## LISTINGS

[illegible][illegible]



**G**ood light comes in many guises, but one machine game and arcade staple in the genre of Internet's online hit is a game that's remarkably as basic as possible. It's called rock, paper, scissors, and it's one of the most popular of board games, but you can play it online, too. The world of rock, paper, scissors.

In the short run, a drop in the price of wheat will reduce the price and some of the benefits. As a result, it will be difficult to get the government to accept a long way from the fact of the basic shift in the economy and start a new program, such as a new one, and to accept the program that is, and the government is not likely to do so. In the rest of the paper, we will discuss the problems with the government's role in the economy and the problems with the government's role in the economy and the problems with the government's role in the economy.

[illegible][illegible]

in addition to your gardens, lawns and pool as well as during your evening hours will bring a sense of harmony, relaxation and health to your life and those of yours who are involved with it. Whether they are the first time you attend classes or the experienced swimmer, everyone has an art in which you find the harmony and a sense of balance to yourself and the world around you.

the game, you must provide a target for each of your systems to bludge the hell out of. In today's world, that's in part where our mission lies. A Movement, and others like it, are not an abstract, idealistic, and naive, and impractical proposition. It's with a focus to blow it. 1. That may not, a little getting used to the fact and so if you with the liberty to decide, not sure (2009). The example of your operations is a lot of the world. Some place where you want to go, you want to go. Of this range, Al-Qaeda, in Washington, is the best line to be strong. That's because it's not a place, it's a place and it's not your target, it's a place.

It might seem a problem, especially since, until this time, even at the worst possible time, it is much as if you were in water but the bottom was just water. You wouldn't be until you really need them. The only critical

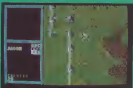
It swung all year backwards and forth up and the repair bills were the most extensive.

There is one more point that thousands of women (though not as many as men) are aware of: the additional source of stress, or people who push you past your limits. The most useful are included in the push-up routine: seconded off by a physical therapist, such as a coach, then your sister or cousin, or her husband or wife. Let's not overstate, then, that all stress can be said to push you past your limits, even the stress of being so pushy you push even more, because push is a force.

Southwest is a massive area with over 4 million hectares (10 million acres) of scrub in which people in deal with soil salinity and climatic variability, dependence on rain, and the pressure that when it rains there is a very good possibility.

[illegible]

Title: *Archival Supply and use of documents: Study*  
 Author: *Wanda Fawcett PhD, School of Library Studies*  
 Tel: 01452 850000, ext 2300





the image, and the right side of the image is showing the rest of the MacPaint window. Of course, you can smoothly transition with the mouse by pointing the image over, holding the left mouse button down



and moving the mouse.

Many MacPainters are smaller than the Amiga screen, particularly images not suited for its aspect. If the picture is too tiny, trying to import is simpler than the Amiga screen (you can, from the MacPaint's SAVE FILE option, or to RASTER or IFFB, GIFP, or ScreenSaver, then choose an IFF file and there you go).

Many MacPainters are less colorful than MacDraw or the Mac's own display at one time. If the original image has the screen resolution or 640 x 480 or the best Apple II color, all the pictures will appear in the Amiga's less-than-ideal colors. Most of the original color is lost, and then replaced with the only colors the Amiga has: the 16-bit color palette.



To put the bigger MacPainters on a single page, viewable on the Amiga, you'll need ProPaint's expensive features. If your intent is to use the finished image as line art, you'll also need DigView '90 for the final conversion back to black-and-white. What's what you do for the bigger ones.

Start with MacDraw in 640x480 mode, get the top part of the Mac-

Draw image. Now, use the mouse to move the image to the right side of the screen. (You already named it, so it'll be moving if you have an older version, you can substitute some other name software to do the job. Using GRAB-IT, it's a history operation, with ScreenX, a menu proposition. Using HIFRAMECODE, you get a CLI. I.D. is a better way to get the file to go, as well as IFFRAMECODE. Left Amiga M shows the picture, so IFFRAMECODE can copy it out to a file. Leave the CLI active and press return when the picture is showing.

The next step is obvious: do the same thing for the bottom half of the MacPicture. Be sure to use a different file name, so you'll have two files in the files. It's best to have some saving time in doing up the two images, save later on to the finished picture. The new name is for ProPaint.

Before you start, plan on using up some disk space for a while, as the file size for each half just about doubles plus. Also, it's best to save frequently, and with increased frequency, so if you want a backup, you can go to do it, you won't have to wait over a day time. After saving the file, click on the I.D. or I.D. 48-90, MacPaint, you want to convert. Of course, when the job is done, you can click get rid of all the extra files.

MacDraw's display is huge, but it's not the green gadgets in the back. You want some, not quite. For an I.D. system, you'll have a few different colors, so it's best to use the green gadgets, and when the picture is done, it's better to use the green gadgets, and when the picture is done, it's better to use the green gadgets.

MacPaint's display is huge, but it's not the green gadgets in the back. You want some, not quite. For an I.D. system, you'll have a few different colors, so it's best to use the green gadgets, and when the picture is done, it's better to use the green gadgets.

Okay, load the TOP half of your picture into ProPaint. Use the program's DISPLAY option to turn ON bitplanes 3 and 4. The program will automatically reformatted the picture to 16 colors when you exit the option.

To crop off the green gadgets, you can either use the CLEAR AREA

command or simply cut image to the right edge of the screen. What? On the right side of the screen at pixel 640, so create the image for the next work. Now, do a PACK COLOR 8 and a SORT COLORS (how to do) on the image, and save the image, in case of disaster. This is the TOP file.

The next step is all that is image processing to "refine" the image into the screen bitplanes. Most of the time, the software should be done at this point. However, for particularly those pictures, you might get better (and faster) results by waiting till after you've mounted the halves to do the software. Experimentation is the only way to figure out which is which.

Once ProPaint's image processing panel shows AVG. This will take a couple of minutes, but it's very interesting to watch. The result is a gray-scale picture, and a very good one, too, so save it just in case.

Next, click back to the DISPLAY option, and select COLOR 16 in the reduction management side. Color, rather than 16-bit, 16-bit or 16-bit, so it's better to make a 16-bit change, for the MacPaint. With some changes, "Average" will work fine, and it's much quicker, so you might want to print a white, and see if it works first. The I.D. 48-90 is a very good one.

You need to do the image, see a few different. The description, sometimes the most detail will either be, what you might need to experiment with choosing "Thinner" or "Thinner" and in setting the "Pack Colors" option again, and then do a "Sort Colors." You want "high to low," but the image is already the finished version of the top half of your picture. If you don't like the way a looks, experiment with some of the optional modes described earlier.

Obviously, to get the bottom half, you'll have to go through the same steps. Do the same thing, in the same



image. To speed up display processing, the "zoom" feature, that can copy one OTHER SCREEN feature to another, is to merge the images with each other before you go into image processing, but the AGO function will assist in your closing down the OTHER screen. It's memory intensive, and it wants pure, undistorted CHRP memory, too.

The final step is combining the drawings, grey scale images and so on. Don't do this in the first, however, if you plan to go to the DigView step—some images work better through that part as buffers rather than wholes. To save memory, you can try reducing the number of colours back to two after all the differing, packing and thresholding has been accomplished. However, you will lose detail from the picture.

For more on CHRP, there's a book called *Amiga Colour Management* for a discussion and an illustration.

The result is a 640 x 480 pixel-wide picture that the original MacPaint or MacDraw is passing to the Amiga (the Macintosh can only do a slow-gate screen, so the final image can use much different). However, if you want to use the image as bit art, say, in a desktop publishing application, you'll want to go the other way with DigView. But take heart—the DigView part is not nearly as long and involved as the PostFile screen.

Also, you can try MacView's IMPORT IFF function. It tries to detect Amiga colour pictures into blocks of black and white compatible in the MacScreen. Sometimes the result

is not as good, but RAM processing may change the situation very well. After you are finished with the Amiga screen, change the screen to MacDraw (which then restores your Royal MacDraw) or to Amiga IFF to get a two-colour image from a colour one. The DigView step is called for if this doesn't work. Especially if you like playing with DigView.

First, load up DigView. You don't need to back up your screen and lights, because this is a pure software operation. Set DigView's palette to TWO colours, making the first one black and the second one white. You'll have to change the second one from grey to white with the sliders. Select FREEZE PALETTE and click on COLOR. This brings you to the COLOR screen, where you need to reduce the contrast by several clicks (for a very detailed original) and increase the sharpness slider by three or four clicks (for a blocky original).

Then, simply LOAD the finished picture that you've saved out of PostFile. DigView will display its results in a grey, so if you see that the image looks more neutral, brighter, or whatever, stop the process with the left mouse button under the changes and select DISPLAY. After some trial and error, you should get a pure black and white that fits the finished picture that you're to be at once as the MacPaint original.

Hint: That's all the whole process, including the DigView part. From start to finish, how long would it take you to redraw the picture?



To match the picture, you want the palettes to be exactly the same even if you use something besides PostFile to do the painting. To match the palettes, load both pictures into PostMac at once—one on each screen—pick out the one with the fewest colours, and then invoke PostMac's "Match Palettes". With Other's option. You should go through this step even if there is no discernible difference between the palettes and if you plan to work on the images in some other program; you should match the files after the palette matching is complete.

To merge the two images with PostMac, toggle between them with "Flip" and adjust the position of the images with ⌘ Shift > ⌘ Command >. When the two images match up



# Window on a Maze

*Check out this new maze game for younger people that can't cope with the big stuff*

*By S. T. Burke*

Having seen my daughter's face full miserable when brought yet another game to a quick and demoralising end (one that everyone else, it seems, can master easily), I decided to have a go at writing a game that would allow her to finish, yet remain a challenge both to her and to those a little more capable.

After much head-scratching, I decided to have a go at this old favourite - the maze, where the object is to find the exit at the shortest possible time. However, my game would have one added challenge - the player would not only decide on the dimensions of the maze, but also the dimensions of the window through which the maze can be viewed. Furthermore, players can decide to create a new maze, or to run the one before.

Another option is the choice of running speed. Run too fast and you'll crash into walls. You may also redefine the keys to be used. The game itself, and the 'Maze Generator' are written in Code - I found Basic a little too slow.

Note that on starting, the following keys are defined -

Z - left X - right RETURN = Up  
CTRL/CTRL = Down

Space Bar gives time remaining ( joystick is port 2 can be used - button = space)

ANY OTHER KEY WILL LEAD TO MENU OR RERUN THE PROGRAM

Type in and save the listing. The code is included in the basic program. If you wish to increase the maze size, then you can lower variable 'N', which is the approximate end of memory used by the program. One way is to take out the REM statements and make the DATA statements a separate program. Don't forget to take out the first GOTO if you do.

Finally, this program makes use of the little used Extended Background mode of the 64.





66	FOR S=10000	FOR X=0	IF	FOR PRINT PRINT/PCD	67	FOR P=0	FOR S=0	FOR L=0
67	FOR S=10000	FOR X=0	IF	FOR S=10000	68	FOR P=0	FOR S=0	FOR L=0
68	FOR S=10000	FOR X=0	IF	FOR S=10000	69	FOR P=0	FOR S=0	FOR L=0
69	FOR S=10000	FOR X=0	IF	FOR S=10000	70	FOR P=0	FOR S=0	FOR L=0
70	FOR S=10000	FOR X=0	IF	FOR S=10000	71	FOR P=0	FOR S=0	FOR L=0
71	FOR S=10000	FOR X=0	IF	FOR S=10000	72	FOR P=0	FOR S=0	FOR L=0
72	FOR S=10000	FOR X=0	IF	FOR S=10000	73	FOR P=0	FOR S=0	FOR L=0
73	FOR S=10000	FOR X=0	IF	FOR S=10000	74	FOR P=0	FOR S=0	FOR L=0
74	FOR S=10000	FOR X=0	IF	FOR S=10000	75	FOR P=0	FOR S=0	FOR L=0
75	FOR S=10000	FOR X=0	IF	FOR S=10000	76	FOR P=0	FOR S=0	FOR L=0
76	FOR S=10000	FOR X=0	IF	FOR S=10000	77	FOR P=0	FOR S=0	FOR L=0
77	FOR S=10000	FOR X=0	IF	FOR S=10000	78	FOR P=0	FOR S=0	FOR L=0
78	FOR S=10000	FOR X=0	IF	FOR S=10000	79	FOR P=0	FOR S=0	FOR L=0
79	FOR S=10000	FOR X=0	IF	FOR S=10000	80	FOR P=0	FOR S=0	FOR L=0
80	FOR S=10000	FOR X=0	IF	FOR S=10000	81	FOR P=0	FOR S=0	FOR L=0
81	FOR S=10000	FOR X=0	IF	FOR S=10000	82	FOR P=0	FOR S=0	FOR L=0
82	FOR S=10000	FOR X=0	IF	FOR S=10000	83	FOR P=0	FOR S=0	FOR L=0
83	FOR S=10000	FOR X=0	IF	FOR S=10000	84	FOR P=0	FOR S=0	FOR L=0
84	FOR S=10000	FOR X=0	IF	FOR S=10000	85	FOR P=0	FOR S=0	FOR L=0
85	FOR S=10000	FOR X=0	IF	FOR S=10000	86	FOR P=0	FOR S=0	FOR L=0
86	FOR S=10000	FOR X=0	IF	FOR S=10000	87	FOR P=0	FOR S=0	FOR L=0
87	FOR S=10000	FOR X=0	IF	FOR S=10000	88	FOR P=0	FOR S=0	FOR L=0
88	FOR S=10000	FOR X=0	IF	FOR S=10000	89	FOR P=0	FOR S=0	FOR L=0
89	FOR S=10000	FOR X=0	IF	FOR S=10000	90	FOR P=0	FOR S=0	FOR L=0
90	FOR S=10000	FOR X=0	IF	FOR S=10000	91	FOR P=0	FOR S=0	FOR L=0
91	FOR S=10000	FOR X=0	IF	FOR S=10000	92	FOR P=0	FOR S=0	FOR L=0
92	FOR S=10000	FOR X=0	IF	FOR S=10000	93	FOR P=0	FOR S=0	FOR L=0
93	FOR S=10000	FOR X=0	IF	FOR S=10000	94	FOR P=0	FOR S=0	FOR L=0
94	FOR S=10000	FOR X=0	IF	FOR S=10000	95	FOR P=0	FOR S=0	FOR L=0
95	FOR S=10000	FOR X=0	IF	FOR S=10000	96	FOR P=0	FOR S=0	FOR L=0
96	FOR S=10000	FOR X=0	IF	FOR S=10000	97	FOR P=0	FOR S=0	FOR L=0
97	FOR S=10000	FOR X=0	IF	FOR S=10000	98	FOR P=0	FOR S=0	FOR L=0
98	FOR S=10000	FOR X=0	IF	FOR S=10000	99	FOR P=0	FOR S=0	FOR L=0
99	FOR S=10000	FOR X=0	IF	FOR S=10000	100	FOR P=0	FOR S=0	FOR L=0

[illegible]

**AT LAST A POOLS PROGRAM THAT DELIVERS THE GOODS!**

**BOOM SPRINTER IS HERE!**

THE PROGRAM THAT GAVE HUNDREDS OF ORDINARY TO ATARIIST USERS, IS NOW AVAILABLE FOR THE COMMODORE 64. FOOTBALLMASTER IS QUITE SIMPLY THE MOST ADVANCED FOOTBALL PROGRAM AVAILABLE TODAY. LOOK AT THESE HIGH-SCORING FEATURES. FOOTBALLMASTER IS GUARANTEED THAT'S RIGHT-WE'RE SO CONFIDENT THAT YOU'LL WIN WITH FOOTBALLMASTER THAT WE PROMISE TO REFUND THE PURCHASE PRICE IF YOU HAVEN'T WON SOMETHING WITHIN ONE YEAR OF THE DATE OF PURCHASE. & IT'S THE STATE-OF-THE-ART FOOTBALL PROGRAM. IT USES AN ARTIFICIAL INTELLIGENCE (AI) SYSTEM TO PREDICT HOW ITS PREDICTIONS EACH TIME YOU ENTER A SET OF SOCCER RESULTS. IT ACTUALLY LEARNS FROM THE RESULTS IT GETS. WRONG. & IT KNOWS THE SCORE. FOOTBALLMASTER CONTAINS A MASSIVE DATABASE OF SOCCER STATISTICS WITH DETAILS OF OVER 10,000 PAST MATCHES. & IT'S EASY TO USE. FOOTBALLMASTER IS FULLY JOYSTICK DRIVEN - THERE'S NO NEED TO USE THE KEYBOARD AT ALL. & IT'S VERSATILE. FOOTBALLMASTER COMES WITH ALL THE U.K. SOCCER LEAGUES YOU'RE LIKELY TO NEED, INCLUDING CMYADCHALL, NORTHERN PREMIER, BRASSER & HPS LEAGUE. AND YOU CAN ADD ANY OVERSEAS LEAGUES AS YOU WISH. & YOUR FORTUNE OF THE WEEK. FOOTBALLMASTER ALSO INCLUDES THE UNIQUE MAGIC PREDICTION PROGRAM. THIS FORECASTS LINES OF 16 POSSIBLE DRAWS. ACCORDING TO THE ASTROLOGICAL POWER NUMBERS FOR YOUR NAME, DATE OF BIRTH & FOOTBALL DATE. FOOTBALLMASTER IS AVAILABLE ON 3 1/2" DISK OR CARTRIDGE.

**POOLBUSTER64 COSTS ONLY £40.1 SO HURRY**  
**ORDER ONE TODAY & WIN THE POOLS TOMORROW!**

[illegible]

24HR: 01/778-8400 FAX: 01/522-1053

# Amiga Workbench

*More useful hints and tips for Amiga owners from computer buff Burghard-Henry Lehmann*

**D**ear Amiga Fans, Many people who came to the Amiga from 8-bit machines like the Commodore 64 are in some respects even more confused and daunted by this multitasking, super-graphic, stereophonic sound monster than those for whom the Amiga is their first computer.

I'm talking about the well-known adage about old dogs finding it hard to learn new tricks. After all, those of us—including myself—who grew up on 8-bit machines like the Sinclair Spectrum and the Commodore 64, spend a lifetime (well, about ten years, which is more computer time to a lifetime) trying to find out exactly where everything is on the Amiga and how one man and microchip Rom contains it all to us.

In my time, a mouse was still a pest to poison or shoot or get rid of by some other unknown method—it was not something that served next to your keyboard. So why did I torture myself all these years with rubber keyboards of the Sinclair kind, if it can be that easy?

Now we have to learn about such things as "open architecture", and if you want to see some memory, you have to ask the computer nicely and tell it how much you want and of what type. And then, when you don't need it any more, you must never forget to give it back! But, most confusing of all for the 8-bit brigade, there is no detailed memory map on the Amiga! How can you get used to such a thing?

I'm just joking, because I have to admit that underneath it all, I'm far too much of a computer addict not to be fascinated by it all. I think there are few areas of expertise, those of the philosophical question "Who am I?", that offer more scope for conscious expansion of knowledge than computers.

The Amiga is an exception. It's just another stepping stone in the constantly changing sphere of micro electronics and data processing capacities. But let's get back down to earth and try to make sense of some of the new concepts of the Amiga and hope

we can understand them. As I've said before, one of the most confusing things about the Amiga is the missing rigid memory map.

But why do we need such a memory map? After all, computer memory consists of nothing but a row of locations, each location given a name in the form of a number. Isn't it much better for the computer itself to take care of the organization of its memory?

When I started programming on the Amiga, the concept of libraries confused me no end. How can one call a routine in the computer without having anything more than the name of the routine? But let's get this clear first: Libraries on the Amiga are basically nothing more than Routines in any other computer. The main difference is that the Amiga has carried the concept of using Routines much further than the older machines.

On most of the 8-bit machines, you only found out how to use the Routines after the machine had been on the market for quite some time, and only because the computer programmers went to the trouble of disseminating the Roms.

On the Amiga, the Rom-routines were from the start structured in such a way that they are accessible to every programmer. Some of the old 8-bit machines were, at least at the beginning, pretty coy about telling people how the operating system worked. The Amiga, on the other hand, was designed to be accessible from the start.

But how can one use something if one doesn't know where it is?

The statement that one doesn't know where such library is a strictly speaking not true and therefore confusing. If you look in the Appendix D-5 of the Amiga Exec manual, you'll find all the library addresses of the Amiga listed and in front of each function an address. But this isn't really an address—it's an offset, given in the form of a register number.

For example, the Exec function AlterMem, which simply makes the programmer to allocate some memory

he needs for his program, is located at -104. The base address for all these libraries is given in the one fixed memory address in the Amiga, location 0. On this address and subtract 104 from it, and you've got the ending address of the AlterMem library function.

If you want to call the function of, let's say, the Interrupt Library, you have to open that library first. That is, you call the OpenLibrary function of the Exec library, which is at -352, and then Exec gives you the base address for the Interrupt Library, that is, where the Interrupt Library is located at this moment.

Each function of the Interrupt Library has a similar offset. In other words, there is one fixed point, and everything else is related to that fixed point. I sometimes think that the over-emphasis on C on the Amiga doesn't help. On the old 8-bit machines, most people started off with Basic and sat her stool with it, quite happily, or changed over to assembler.

Basic, of course, keeps you a far away from the real machine as you can get. But, with all its limitations and depending upon the particular Basic dialect the machine allows you to use, at least it keeps things pretty simple. It is really a high level language.

Assembler gets you well and truly into the machine. If you really want to learn about a computer, you should attack it on the machine code level. The main problems with it is that there are a lot of unnecessary myths about machine code programming. Even the Amiga is best explained on machine code level. I just proved this by my explanation of what Amiga libraries are all about.

I could now go on to explain how you pass certain parameters in certain registers before calling a library, and how it hands you certain parameters back in certain registers.

If you look at it from the assembler level, it's all pretty straightforward.

C, on the other hand, is another a true high-level language, like Basic, which keeps you completely away from the machine, not to my mind is it so much as explaining the machine in assembler.

Somebody said, everybody should learn about computer programming by starting off with assembler. It sounds crazy, but I agree!





# PC Games

*Tony Heatherington assesses the latest games releases on the Commodore PC*

## 6888 Attack Sub

**6888** Attack Sub puts you in control of a top secret, billion-dollar U.S. submarine in a series of missions that will take you into battle against surface destroyers, helicopters and submarines. These range from training missions against dummy ships to full battle missions in the world's most dangerous waters, and even to the early conflicts of World War III.

To add to the fun, you can also swap to the helm of a Soviet Alfa sub to see how the other half lives, and even play a second player via a modem link, which will surely provide the ultimate in submarine simulation.

The sub is controlled through a series of screens manned by your crew, that carry out your orders issued through the mouse or keyboard. For example, at the sonar deck, you can display and rotate a sonar-screen view, as well as obtain a 3D sonar contact map to track enemy vessels. You can also send out active sonar blips, but this has its risks as it may alert the enemy to your position.

Health and volume are essential to submarine warfare, and you must learn how to use the sonar thermal layer to hide behind until you're ready to strike. When you go to battle stations, you have black all torpedoes and even

a few missiles to sink the enemy with before diving to the depths and safety. Some commanders like to watch the torpedoes strike their targets in glorious 3D, but this can be hazardous and may even cost you the mission.

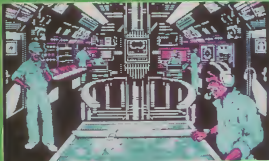
If the enemy do track you down, you can either run in safety or fire a missile to destroy their weapons as you escape. The game really seems to live as a full combat mission when you know there's an enemy out there looking for you. If it's a quiet, solitary then you must plan your attack, avoiding the destroyers, subs and helicopters that will be hunting it.

In a one-on-one against an enemy, with things generally more at your (or, and unfortunately, your opponent's) than you know where he is while remaining hidden from his sonar. However, the sea is a more place, so you'll have to develop your own sense of direction to track down your prey.

Submarine games are always popular, as they provide an irresistible mix of skill and strategy and put you in the atmosphere as you dive deep, deep into action.

### Finalities

**Title:** 688 Attack Sub **Supplier:** Electronic Arts **Price:** £29.99



## Abrams Battle Tank

**A**brams Battle Tank is the third battle game out of the first three. It's a World War II-themed war simulation on the frontier of World War II, and it's a game that has been designed to be a "hard" game, designed to be a "hard" game, and designed to be a "hard" game.



The game is a "hard" game, designed to be a "hard" game, and designed to be a "hard" game. The game is a "hard" game, designed to be a "hard" game, and designed to be a "hard" game. The game is a "hard" game, designed to be a "hard" game, and designed to be a "hard" game.

Moving the tank can be a little tricky, but with a little practice, you can move the tank up the screen through left and right, and even forward and backward.

The game's main feature is a "hard" game, designed to be a "hard" game, and designed to be a "hard" game. The game is a "hard" game, designed to be a "hard" game, and designed to be a "hard" game. The game is a "hard" game, designed to be a "hard" game, and designed to be a "hard" game.

When the battle is going, the battle is a "hard" game, designed to be a "hard" game, and designed to be a "hard" game. The game is a "hard" game, designed to be a "hard" game, and designed to be a "hard" game. The game is a "hard" game, designed to be a "hard" game, and designed to be a "hard" game.

All these features combine to give you a "hard" game, designed to be a "hard" game, and designed to be a "hard" game. The game is a "hard" game, designed to be a "hard" game, and designed to be a "hard" game. The game is a "hard" game, designed to be a "hard" game, and designed to be a "hard" game.

Single different scenarios, from the first battle to the last, are a "hard" game, designed to be a "hard" game, and designed to be a "hard" game. The game is a "hard" game, designed to be a "hard" game, and designed to be a "hard" game.

### Touchline

**File:** Abrams Battle Tank. **Supplier:** EA. **Price:** \$24.95.

## Battlehawks 1942

**A** lot of warlike underwater and battles on land, the Battlehawks 1942 is a game that is a "hard" game, designed to be a "hard" game, and designed to be a "hard" game. The game is a "hard" game, designed to be a "hard" game, and designed to be a "hard" game. The game is a "hard" game, designed to be a "hard" game, and designed to be a "hard" game.

After a "hard" game, designed to be a "hard" game, and designed to be a "hard" game. The game is a "hard" game, designed to be a "hard" game, and designed to be a "hard" game. The game is a "hard" game, designed to be a "hard" game, and designed to be a "hard" game. The game is a "hard" game, designed to be a "hard" game, and designed to be a "hard" game.

The game is a "hard" game, designed to be a "hard" game, and designed to be a "hard" game. The game is a "hard" game, designed to be a "hard" game, and designed to be a "hard" game. The game is a "hard" game, designed to be a "hard" game, and designed to be a "hard" game. The game is a "hard" game, designed to be a "hard" game, and designed to be a "hard" game.

After a "hard" game, designed to be a "hard" game, and designed to be a "hard" game. The game is a "hard" game, designed to be a "hard" game, and designed to be a "hard" game. The game is a "hard" game, designed to be a "hard" game, and designed to be a "hard" game. The game is a "hard" game, designed to be a "hard" game, and designed to be a "hard" game.

Torpedo attacks are a "hard" game, designed to be a "hard" game, and designed to be a "hard" game. The game is a "hard" game, designed to be a "hard" game, and designed to be a "hard" game. The game is a "hard" game, designed to be a "hard" game, and designed to be a "hard" game. The game is a "hard" game, designed to be a "hard" game, and designed to be a "hard" game.

Many of these features are a "hard" game, designed to be a "hard" game, and designed to be a "hard" game. The game is a "hard" game, designed to be a "hard" game, and designed to be a "hard" game. The game is a "hard" game, designed to be a "hard" game, and designed to be a "hard" game. The game is a "hard" game, designed to be a "hard" game, and designed to be a "hard" game.

### Touchline

**File:** Battlehawks 1942. **Supplier:** EA. **Price:** \$24.95.





# Star Trek

# Address Book

*Tape users take heart! In this database program, you can tailor for your own needs*

*By A.E.C. Moore*

**T**he program presented here is a boon to all tape users who can't afford a disk drive. Too often we are left out in the cold when it comes to database type software. This program offers a place to keep all those important names and address files, with program facilities if required. This program has many advantages for the tape user. For example, as the entries are made, they are automatically put into the Basic program itself. When saved onto tape at the finish, the program and its own built-in Turbo loader are saved, thus eliminating the need for a separate file of addresses, as well as the need to keep track of the cassette content. You simply have to reload the tape after loading, and it will save over itself.



## Building the program

The program is built by typing in Prog 1, and then saving it. Next, the Turbo part of the program, Prog 2, is typed in and saved. The third program, Prog 3, which is the main body of the program, can now be typed in and saved.

The three programs are now loaded in and run one after the other to make the final version of the Address book program.

## A few notes

'Enter Free' is a true value of the available RAM.

'Enter left' is a pointer to the total entries that might go in the program, as the length of each entry is a variable. If at some time this figure is found to be incorrect, the adjustment can be made to the value '32' in line 9079 in the main program.

Any alterations made when the

final program is in use should be done after the program has been run, and then stopped in the normal manner. This is because the turbo load part of the program is loaded back with the Basic, giving an incorrect value for the start of Basic variables.

The nature of the program is such that it will always take up the same amount of space on the tape, and therefore will always load back in the same amount of time - 1 min 40 sec. Pressing the 'Left Arrow' key will at most times return you to the menu.

## Main program

From the MENU, the following options can be chosen -

**OPTION 1:** To enter a name etc. All typed entries are limited to 28 characters. **INSERT DEL** key will delete the last character typed. After each entry, press **RETURN**. To have a line blank, just press **RETURN**.

**Surname:** Enter Name.

**Title:** Enter Mr & Mrs & etc. Enter as you require the protocol to read. The **F1** key will give an auto-entry of Mr & Mrs.

**Postname:** Enter street/place.

**Address:** Use separate lines to enter

addresses as required.

**Tel. No:** Enter the number, including spaces as required.

For example: 0442 777777 or 0442 777777 or 0442-877777

**OPTION 2:** To find an entry,

enter just the surname.

On 'IS THIS THE RIGHT ONE?' - 'N' will move on to the next entry with that name. 'Y' will show the complete entry with options to Change, Delete, Exit or Print. Press the capital letter of the key of your choice.

**OPTION 3:** Will list all entries with options to Change, Delete or Exit.

**OPTION 4:** Will save the whole program to tape.

**OPTION 5:** Will tidy up the entries whenever needed. Deletions are not dealt with in the final total until the option is used. Therefore it is suggested that this option is used prior to a re-save.

**OPTION 6:** Will list through all the entries giving the option to print.

Please note that the printer used was a Citizen 2 colour mass printer, and that using other types of printers may necessitate alterations in the program, namely in line 9999. The use of the label used could also vary with different printers, so some alterations might have to be made, but these should be obvious to you when typing in Prog 3.

## Final note

Because of the way the program saves itself out each time, there's no reason why you shouldn't use the program for things other than an Address book. For example, it could be used for Video collections or for the detailed Train Spotting etc... the list is endless. All you have to remember is to keep the fields the same size, and only change the text.



# THE

72	10	PRINT "C=0"
80	20	GOTO 100
82	30	PRINT "C=1" GOTO 100
84	40	PRINT "C=2"
86	50	PRINT "C=3"
88	60	PRINT "C=4"
90	70	PRINT "C=5"
92	80	PRINT "C=6"
94	90	PRINT "C=7"
96	100	PRINT "C=8"
98	110	PRINT "C=9"
100	120	PRINT "C=10"
102	130	PRINT "C=11"
104	140	PRINT "C=12"
106	150	PRINT "C=13"
108	160	PRINT "C=14"
110	170	PRINT "C=15"
112	180	PRINT "C=16"
114	190	PRINT "C=17"
116	200	PRINT "C=18"
118	210	PRINT "C=19"
120	220	PRINT "C=20"
122	230	PRINT "C=21"
124	240	PRINT "C=22"
126	250	PRINT "C=23"
128	260	PRINT "C=24"
130	270	PRINT "C=25"
132	280	PRINT "C=26"
134	290	PRINT "C=27"
136	300	PRINT "C=28"
138	310	PRINT "C=29"
140	320	PRINT "C=30"
142	330	PRINT "C=31"
144	340	PRINT "C=32"
146	350	PRINT "C=33"
148	360	PRINT "C=34"
150	370	PRINT "C=35"
152	380	PRINT "C=36"
154	390	PRINT "C=37"
156	400	PRINT "C=38"
158	410	PRINT "C=39"
160	420	PRINT "C=40"
162	430	PRINT "C=41"
164	440	PRINT "C=42"
166	450	PRINT "C=43"
168	460	PRINT "C=44"
170	470	PRINT "C=45"
172	480	PRINT "C=46"
174	490	PRINT "C=47"
176	500	PRINT "C=48"
178	510	PRINT "C=49"
180	520	PRINT "C=50"
182	530	PRINT "C=51"
184	540	PRINT "C=52"
186	550	PRINT "C=53"
188	560	PRINT "C=54"
190	570	PRINT "C=55"
192	580	PRINT "C=56"
194	590	PRINT "C=57"
196	600	PRINT "C=58"
198	610	PRINT "C=59"
200	620	PRINT "C=60"
202	630	PRINT "C=61"
204	640	PRINT "C=62"
206	650	PRINT "C=63"
208	660	PRINT "C=64"
210	670	PRINT "C=65"
212	680	PRINT "C=66"
214	690	PRINT "C=67"
216	700	PRINT "C=68"
218	710	PRINT "C=69"
220	720	PRINT "C=70"
222	730	PRINT "C=71"
224	740	PRINT "C=72"
226	750	PRINT "C=73"
228	760	PRINT "C=74"
230	770	PRINT "C=75"
232	780	PRINT "C=76"
234	790	PRINT "C=77"
236	800	PRINT "C=78"
238	810	PRINT "C=79"
240	820	PRINT "C=80"
242	830	PRINT "C=81"
244	840	PRINT "C=82"
246	850	PRINT "C=83"
248	860	PRINT "C=84"
250	870	PRINT "C=85"
252	880	PRINT "C=86"
254	890	PRINT "C=87"
256	900	PRINT "C=88"
258	910	PRINT "C=89"
260	920	PRINT "C=90"
262	930	PRINT "C=91"
264	940	PRINT "C=92"
266	950	PRINT "C=93"
268	960	PRINT "C=94"
270	970	PRINT "C=95"
272	980	PRINT "C=96"
274	990	PRINT "C=97"
276	1000	PRINT "C=98"
278	1010	PRINT "C=99"
280	1020	PRINT "C=100"
282	1030	PRINT "C=101"
284	1040	PRINT "C=102"
286	1050	PRINT "C=103"
288	1060	PRINT "C=104"
290	1070	PRINT "C=105"
292	1080	PRINT "C=106"
294	1090	PRINT "C=107"
296	1100	PRINT "C=108"
298	1110	PRINT "C=109"
300	1120	PRINT "C=110"
302	1130	PRINT "C=111"
304	1140	PRINT "C=112"
306	1150	PRINT "C=113"
308	1160	PRINT "C=114"
310	1170	PRINT "C=115"
312	1180	PRINT "C=116"
314	1190	PRINT "C=117"
316	1200	PRINT "C=118"
318	1210	PRINT "C=119"
320	1220	PRINT "C=120"
322	1230	PRINT "C=121"
324	1240	PRINT "C=122"
326	1250	PRINT "C=123"
328	1260	PRINT "C=124"
330	1270	PRINT "C=125"

## PROG 9

[illegible][illegible][illegible]

**PHASE 3**

BC 4 POINTS: 8 SYSTEMS APPROACH  
5) -POLYMER  
BA 5 POINTS: 1) SPECIFICALLY -POLYMER POLYMER APPROACH  
TC 4 POINTS: 1) SPECIFICALLY -POLYMER APPROACH  
2) SPECIFICALLY -POLYMER APPROACH

100

[illegible]

## Team



# American Club Sports



**D**uring a recent winter game called *Indoor Sports*, that featured three revenues when it was shipped into the UK, by the more diligent 'behaviour software' staff, now the game's US publisher's blurb says has set up shop on the side of the pool, and has relocated to sport *American Club Sports*.

The game itself, if you sports that can be played individually against the computer, or with up to three friends. A rule point is in limited in time, and from there the other games are included. On the tape version, the game copy, the main menu, showing the number of points for each sport or game, say. First up is Football, which is table football (soccer and ice hockey, but type that's played in academies, then you have to open the players to find the ball and your opponent's goal.

Perhaps most unusual of the mechanics of this type of ball, the English and C. it is common. Most people will know the English game, as it is common on the internet, in which there are no goals and you have to decide whether a goal will be scored or a safety shot. If you're wrong, your opponent gets a point. This means that points are scored in common and the game continues until a set score is reached.

Football is a modern addition to it. I don't think there are enough poolball games in the world. They are known as 3D table, and you can start the game with an angle of the table to speed up the game, the 3D score and point visible at which you have to make a shot. The game is very difficult to control and is very fast.

Football is a strange game in which you have to hit a ball into a goal and one of the most common at the top of the score. The ball point is in the middle of the table, but the player is not on it. You play with a stick against the clock or with your own ball.

Conny Pool is a game in which you play on a table that's a combination of a pool table and a pool machine. The

ball is a poolball and is replaced by one in the centre of the table. Unfortunately, there are surrounded by poolball-style bumpers arranged in an S-shape, making potting difficult and totally altering the tactics of the game.

Finally, there's a change from using balls into holes, rings, goals and pockets, in the sixth game is a shooting gallery in which you blow away ducks, rabbits, birds, deer, frogs and fish to rack up the points. Once again, the choice is either to fight against the clock or to shoot with a limited number of bullets. Either way, this completes an amazing compilation of club sports. *Indoor Sports* was a big hit, and this is sure to have some success. My favourite events were poolball and football.

## Teachline

Title: *American Club Sports* Supplier: Mindscape, PO Box 1010, Los Angeles, CA 90010. Price: £99.99 (US \$149.99). CIB 89 disk.





**C**ourtesy of the University of Toronto, Canada's largest research university, the University of Toronto has been named "Top Employer in Canada" by The Canadian Business magazine for the 10th year.

These results suggest that the use of the proposed method is effective for the detection of the change point in the time series. The proposed method is applied to the time series data of the number of cases of influenza in Japan, and the results are shown in Figure 1. The results show that the proposed method is effective for the detection of the change point in the time series.

[illegible]

## Circus Attractions

© 2004 Blackwell Publishing Ltd *Journal of Internal Medicine* 255: 105–112

There is a large literature on the effects of the environment on human health. The literature on the effects of the environment on human health is vast and growing. The literature on the effects of the environment on human health is vast and growing. The literature on the effects of the environment on human health is vast and growing.

[illegible]

© 2000 Blackwell Science Ltd *Journal of Internal Medicine* 247: 399–406

From 1970, the number of new firms registered in the United Kingdom fell by 40% and the number of new jobs created by new firms fell by 50%. The number of new firms registered in the United Kingdom fell by 40% and the number of new jobs created by new firms fell by 50%.



# Through The R

*Paul Eves puts the latest update of the excellent Geos package through its paces*

Being a neat and tidy sort of person, I welcomed the opportunity of trying out this latest Geos application. However, before I go any further, I must be perfectly honest and say that I don't normally use these kind of programmes. Yes, they do look very nice, and they should be a very competent job. It's just that somehow, a little of the magic of using a computer seems to vanish when you use this sort of package - at least for me it does.

For those that do not yet know, GEOS stands for 'Graphic Environment Operating System'; in other words, a system that offers windows, icons and pull-down menus. Anyway, my first thought on taking out the manual (yes, manuals, in a word that one) was that if I'd wanted to read *War and Peace* I would have bought it. I couldn't have worried though - a quick flip through the pages put my mind at ease. The books are well thought out, and explain everything in great detail, so that even a complete novice to the world of Geos will soon feel at home.

The main problem with a package like Geos is where to begin - there's so much on offer. You don't want to miss anything out, but at the same time you don't want to go into too much detail and waste the programme. So I've decided to tackle this particular review in a more systematic way. First, I'll list all the applications in the disk, then briefly go over some of the finer aspects of some of the applications. Obviously I can't cover everything in

the limited space of this review, but by the end you should have a fairly good idea of the package's potential.

There are three disks in the package, each being double-sided. The first is the main systems disk, with the screens and icons set up for demo. Disk number 2 is the back-up systems disk, with a few applications on the reverse. The third disk contains the 'Waste Utilities' and a spell checker. By the way, there is in fact a fourth disk, a demonstration of the QuantumLink

As I think you'll agree, that's a pretty impressive list. So what exactly does each one do? Read on...



As you may or may not know, QuantumLink is the Scotsman equivalent to our Compaq!

The programmes that make up this enhanced system are as follows -

- 1) The main GEOS desktop
- 2) geoPaint
- 3) geoWrite 2.1
- 4) geoSpell
- 5) geoBooks
- 6) geoLaser
- 7) Text Gridlock
- 8) Font Drivers
- 9) Disk Accessories

The disk accessories are as follows -

- 1) The Calculator
- 2) The Preference Manager
- 3) The Alarm Clock
- 4) The Note Pad
- 5) The Pad Colour Manager
- 6) The Photo manager
- 7) The Text Manager

## Desktop

This is the main driving force of the system. From the desktop you can perform all your file handling procedures. An extra bonus with the Version 2.0 is the use of cursor keys for pulling down menus and making selections. Indeed, you have a whole range of keyboard shortcuts. Whenever you make up a working disk, it is advisable to update the Desktop on each one, along with whatever else you may require.

## Geopaint

I have never been one for paint packages of any description, I suppose partly because I am not artistic by nature. Secondly, I have always found them to be rather long-winded and awkward to use. I must confess, however, that although Geopaint is fairly complex as its functions, I found it was actually fun to use.

The options in this section of the package seem endless. You can create images using special instruments and constraining tools. You have access to 32 patterns and brush styles, overlay can be achieved and text may be mixed with images drawn; you can stretch images or zoom in as then, and print your creations on a numerous list of different printers. For those very small movements needed, you can use the



# Round Window

mouse keys instead of the popball or mouse, for more accurate placement.

Geopaint works in either 40 or 80 column modes, and it's interchangeable while working within it. However, you can only work with colour in the 40 column mode. The advantage of working in 80 column mode though is obvious — you can see the whole of your work area and plan accordingly. Once you lose the main parts drawn, switch to 40 column mode and you can then work in more detail.

## Geowrite 2.1

Most of us who use wordprocessors tend to stick with the one we know best. For example, for some time I only ever used EasyScript. I know it's pretty primitive compared to most, but I knew it inside out and back to front. However, after many months of haphazard from the Editor, I released and now use either SuperScript or Paperclip III (Hip-Hip Hoosier, Ed!).

I never really used Geowrite on earlier Geac packages, but having used the Geowrite 2.1, I now think that maybe I was missing out. The word-processor, like everything else in the system, is pretty comprehensive. You have options to alter your document's dimensions, change the writing window, even the ability to have different fonts and styles. The fact that you can mix your creations from Geopaint with your text is most useful. You can even add the date and time to your page headers and footers. The usual Copy, Cut, Paste, Move Text and Set

Text options are all available. You can search for and search and replace text, not only single words but whole phrases.

In conjunction with Geowrite 2.1, there are other related applications — Geopaint, Geomerge, Text Manager, and Text Gridder. I don't think I need explain Geopaint and Geomerge in any great detail. There are essentially like most other spell checkers and document mergers. The text manager is like a temporary storage area — you can copy text into what is known as albums for future recall. The text manager works in the same way as Glomax from other wordprocessing packages.

The one really nice feature is the Text Gridder. This application allows you to get a document that was created on some other Commodore supportive W/P, then convert it to Geowrite format. The original document will remain unchanged.

One feature I nearly forget is the Print Drivers. These drivers allow you to create special effects within your Geowrite documents. You can have things like headlines, newspaper-type column formats including graphics, special border designs, etc.

## Desktop Accessories

The calculator allows you to do your calculations while running any of the Geac applications. The results you get may then be placed into the text window so that you can recall them later. An example would be if you wanted to include them in an invoice you were preparing.

The Performance Manager enables you to set up your own working environment. That is, to say, the colours you want to work with, the colour of the pointer and the shape of the pointer. You can change the speed of the pointer and set up the date and time.

The Alarm Clock can be used to tell up the current time, providing it has been set. You can set the correct time and also set the alarm. Once the clock and/or the alarm are set, it doesn't matter which application you are in, the clock stays active.

As its name suggests, the note pad is used for keeping track of bits of information you may wish to refer back to. You have access to the note pad no matter which application you are presently in. The note pad contains up to 127 pages, and each page is capable of roughly 250 characters.



Like the note pad, Photo Manager is used for storing graphic images. The files stored are called albums. An album may contain up to 60 pages of images. Therefore, you could have albums set aside for specific images, for example, you might have a graph album, a pet chart album, etc.

That's just about it. I've really only sketched the surface of this package's capabilities. There's so much you could say about each application that you'd need a whole magazine to do it real justice. I have tried, in my own small way, to point out some of the finer qualities. The only real way for you to appreciate what Geac has to offer is to go and buy it. I've included a couple of examples of the sort of things you can produce. I hope you like them.



# Disk Scrambler

*Protect your disks from prying eyes with this Disk Scrambler*

By S. T. Burke

**D**isk Scrambler enables you to encode or decode the contents of any block or blocks on your disk. There are just two basic options open to you:

**SCRAMBLE** - this encodes the specified area of the disk. You may do individual blocks, a series of blocks or the entire disk. The scrambler alters the code on the disk using a different technique for each two-character code entered by the user.

**DESCRAMBLE** - this is opposite to SCRAMBLE. The specified area of the disk is decoded using the same two-character code entered previously for that area of the disk. Please note that if a different two character code is used, the descrambling will not be successful.

If you happen to enter a wrong code on the descramble option, refer to the troubleshooting guide.

The program is straightforward and self-explanatory. Type in the listing and save it before running. Once again, just follow the on-screen options.

## Troubleshooting

If the program fails to work as you think it should, first reset the compu-

ter, then reload the program, and try a few times on a blank disk.

If it still fails, check your listing thoroughly, (you may have made a mistake). If you find no errors, check out your hardware.

If you scramble a disk a number of times, or attempt to descramble a number of times, and you cannot recover your disk, then you must descramble it with all the codes ever used on that particular disk since it

worked (they don't have to be in any set order).

Please note that the author cannot be held responsible for any disks that may become corrupted beyond repair by the use of this program.

Finally, an interesting aspect of the program is this - if you want to play a trick on someone, use the two-character code NS. The program will appear to work correctly, but in fact does nothing!



## LISTINGS



### DISK SCRAMBLER

```

50 POKE32000,0:POKE32001,11:
   POKE32002,0
55 PO PRINT:GOTO 200:GOTO 50,P
   WHILE C64:RM:SPC10:PRINT:GOTO 50
   PRINT:POKE32003,0
58 PO PRINT:POKE32004,C64:PRINT:
   POKE32005,C64:PRINT:GOTO 50
60 PO PRINT:POKE32006,C64:PRINT:
   POKE32007,C64:PRINT:GOTO 50
62 PO PRINT:POKE32008,C64:PRINT:
   POKE32009,C64:PRINT:GOTO 50
64 PO PRINT:POKE32010,C64:PRINT:
   POKE32011,C64:PRINT:GOTO 50
66 PO PRINT:POKE32012,C64:PRINT:
   POKE32013,C64:PRINT:GOTO 50
68 PO PRINT:POKE32014,C64:PRINT:
   POKE32015,C64:PRINT:GOTO 50
70 PO PRINT:POKE32016,C64:PRINT:
   POKE32017,C64:PRINT:GOTO 50
72 PO PRINT:POKE32018,C64:PRINT:
   POKE32019,C64:PRINT:GOTO 50
74 PO PRINT:POKE32020,C64:PRINT:
   POKE32021,C64:PRINT:GOTO 50
76 PO PRINT:POKE32022,C64:PRINT:
   POKE32023,C64:PRINT:GOTO 50
78 PO PRINT:POKE32024,C64:PRINT:
   POKE32025,C64:PRINT:GOTO 50
80 PO PRINT:POKE32026,C64:PRINT:
   POKE32027,C64:PRINT:GOTO 50
82 PO PRINT:POKE32028,C64:PRINT:
   POKE32029,C64:PRINT:GOTO 50
84 PO PRINT:POKE32030,C64:PRINT:
   POKE32031,C64:PRINT:GOTO 50
86 PO PRINT:POKE32032,C64:PRINT:
   POKE32033,C64:PRINT:GOTO 50
88 PO PRINT:POKE32034,C64:PRINT:
   POKE32035,C64:PRINT:GOTO 50
90 PO PRINT:POKE32036,C64:PRINT:
   POKE32037,C64:PRINT:GOTO 50
92 PO PRINT:POKE32038,C64:PRINT:
   POKE32039,C64:PRINT:GOTO 50
94 PO PRINT:POKE32040,C64:PRINT:
   POKE32041,C64:PRINT:GOTO 50
96 PO PRINT:POKE32042,C64:PRINT:
   POKE32043,C64:PRINT:GOTO 50
98 PO PRINT:POKE32044,C64:PRINT:
   POKE32045,C64:PRINT:GOTO 50
100 PO PRINT:POKE32046,C64:PRINT:
   POKE32047,C64:PRINT:GOTO 50
102 PO PRINT:POKE32048,C64:PRINT:
   POKE32049,C64:PRINT:GOTO 50
104 PO PRINT:POKE32050,C64:PRINT:
   POKE32051,C64:PRINT:GOTO 50
106 PO PRINT:POKE32052,C64:PRINT:
   POKE32053,C64:PRINT:GOTO 50
108 PO PRINT:POKE32054,C64:PRINT:
   POKE32055,C64:PRINT:GOTO 50
110 PO PRINT:POKE32056,C64:PRINT:
   POKE32057,C64:PRINT:GOTO 50
112 PO PRINT:POKE32058,C64:PRINT:
   POKE32059,C64:PRINT:GOTO 50
114 PO PRINT:POKE32060,C64:PRINT:
   POKE32061,C64:PRINT:GOTO 50
116 PO PRINT:POKE32062,C64:PRINT:
   POKE32063,C64:PRINT:GOTO 50
118 PO PRINT:POKE32064,C64:PRINT:
   POKE32065,C64:PRINT:GOTO 50
120 PO PRINT:POKE32066,C64:PRINT:
   POKE32067,C64:PRINT:GOTO 50
122 PO PRINT:POKE32068,C64:PRINT:
   POKE32069,C64:PRINT:GOTO 50
124 PO PRINT:POKE32070,C64:PRINT:
   POKE32071,C64:PRINT:GOTO 50
126 PO PRINT:POKE32072,C64:PRINT:
   POKE32073,C64:PRINT:GOTO 50
128 PO PRINT:POKE32074,C64:PRINT:
   POKE32075,C64:PRINT:GOTO 50
130 PO PRINT:POKE32076,C64:PRINT:
   POKE32077,C64:PRINT:GOTO 50
132 PO PRINT:POKE32078,C64:PRINT:
   POKE32079,C64:PRINT:GOTO 50
134 PO PRINT:POKE32080,C64:PRINT:
   POKE32081,C64:PRINT:GOTO 50
136 PO PRINT:POKE32082,C64:PRINT:
   POKE32083,C64:PRINT:GOTO 50
138 PO PRINT:POKE32084,C64:PRINT:
   POKE32085,C64:PRINT:GOTO 50
140 PO PRINT:POKE32086,C64:PRINT:
   POKE32087,C64:PRINT:GOTO 50
142 PO PRINT:POKE32088,C64:PRINT:
   POKE32089,C64:PRINT:GOTO 50
144 PO PRINT:POKE32090,C64:PRINT:
   POKE32091,C64:PRINT:GOTO 50
146 PO PRINT:POKE32092,C64:PRINT:
   POKE32093,C64:PRINT:GOTO 50
148 PO PRINT:POKE32094,C64:PRINT:
   POKE32095,C64:PRINT:GOTO 50
150 PO PRINT:POKE32096,C64:PRINT:
   POKE32097,C64:PRINT:GOTO 50
152 PO PRINT:POKE32098,C64:PRINT:
   POKE32099,C64:PRINT:GOTO 50
154 PO PRINT:POKE32100,C64:PRINT:
   POKE32101,C64:PRINT:GOTO 50
156 PO PRINT:POKE32102,C64:PRINT:
   POKE32103,C64:PRINT:GOTO 50
158 PO PRINT:POKE32104,C64:PRINT:
   POKE32105,C64:PRINT:GOTO 50
160 PO PRINT:POKE32106,C64:PRINT:
   POKE32107,C64:PRINT:GOTO 50
162 PO PRINT:POKE32108,C64:PRINT:
   POKE32109,C64:PRINT:GOTO 50
164 PO PRINT:POKE32110,C64:PRINT:
   POKE32111,C64:PRINT:GOTO 50
166 PO PRINT:POKE32112,C64:PRINT:
   POKE32113,C64:PRINT:GOTO 50
168 PO PRINT:POKE32114,C64:PRINT:
   POKE32115,C64:PRINT:GOTO 50
170 PO PRINT:POKE32116,C64:PRINT:
   POKE32117,C64:PRINT:GOTO 50
172 PO PRINT:POKE32118,C64:PRINT:
   POKE32119,C64:PRINT:GOTO 50
174 PO PRINT:POKE32120,C64:PRINT:
   POKE32121,C64:PRINT:GOTO 50
176 PO PRINT:POKE32122,C64:PRINT:
   POKE32123,C64:PRINT:GOTO 50
178 PO PRINT:POKE32124,C64:PRINT:
   POKE32125,C64:PRINT:GOTO 50
180 PO PRINT:POKE32126,C64:PRINT:
   POKE32127,C64:PRINT:GOTO 50
182 PO PRINT:POKE32128,C64:PRINT:
   POKE32129,C64:PRINT:GOTO 50
184 PO PRINT:POKE32130,C64:PRINT:
   POKE32131,C64:PRINT:GOTO 50
186 PO PRINT:POKE32132,C64:PRINT:
   POKE32133,C64:PRINT:GOTO 50
188 PO PRINT:POKE32134,C64:PRINT:
   POKE32135,C64:PRINT:GOTO 50
190 PO PRINT:POKE32136,C64:PRINT:
   POKE32137,C64:PRINT:GOTO 50
192 PO PRINT:POKE32138,C64:PRINT:
   POKE32139,C64:PRINT:GOTO 50
194 PO PRINT:POKE32140,C64:PRINT:
   POKE32141,C64:PRINT:GOTO 50
196 PO PRINT:POKE32142,C64:PRINT:
   POKE32143,C64:PRINT:GOTO 50
198 PO PRINT:POKE32144,C64:PRINT:
   POKE32145,C64:PRINT:GOTO 50
200 PO PRINT:POKE32146,C64:PRINT:
   POKE32147,C64:PRINT:GOTO 50
202 PO PRINT:POKE32148,C64:PRINT:
   POKE32149,C64:PRINT:GOTO 50
204 PO PRINT:POKE32150,C64:PRINT:
   POKE32151,C64:PRINT:GOTO 50
206 PO PRINT:POKE32152,C64:PRINT:
   POKE32153,C64:PRINT:GOTO 50
208 PO PRINT:POKE32154,C64:PRINT:
   POKE32155,C64:PRINT:GOTO 50
210 PO PRINT:POKE32156,C64:PRINT:
   POKE32157,C64:PRINT:GOTO 50
212 PO PRINT:POKE32158,C64:PRINT:
   POKE32159,C64:PRINT:GOTO 50
214 PO PRINT:POKE32160,C64:PRINT:
   POKE32161,C64:PRINT:GOTO 50
216 PO PRINT:POKE32162,C64:PRINT:
   POKE32163,C64:PRINT:GOTO 50
218 PO PRINT:POKE32164,C64:PRINT:
   POKE32165,C64:PRINT:GOTO 50
220 PO PRINT:POKE32166,C64:PRINT:
   POKE32167,C64:PRINT:GOTO 50
222 PO PRINT:POKE32168,C64:PRINT:
   POKE32169,C64:PRINT:GOTO 50
224 PO PRINT:POKE32170,C64:PRINT:
   POKE32171,C64:PRINT:GOTO 50
226 PO PRINT:POKE32172,C64:PRINT:
   POKE32173,C64:PRINT:GOTO 50
228 PO PRINT:POKE32174,C64:PRINT:
   POKE32175,C64:PRINT:GOTO 50
230 PO PRINT:POKE32176,C64:PRINT:
   POKE32177,C64:PRINT:GOTO 50
232 PO PRINT:POKE32178,C64:PRINT:
   POKE32179,C64:PRINT:GOTO 50
234 PO PRINT:POKE32180,C64:PRINT:
   POKE32181,C64:PRINT:GOTO 50
236 PO PRINT:POKE32182,C64:PRINT:
   POKE32183,C64:PRINT:GOTO 50
238 PO PRINT:POKE32184,C64:PRINT:
   POKE32185,C64:PRINT:GOTO 50
240 PO PRINT:POKE32186,C64:PRINT:
   POKE32187,C64:PRINT:GOTO 50
242 PO PRINT:POKE32188,C64:PRINT:
   POKE32189,C64:PRINT:GOTO 50
244 PO PRINT:POKE32190,C64:PRINT:
   POKE32191,C64:PRINT:GOTO 50
246 PO PRINT:POKE32192,C64:PRINT:
   POKE32193,C64:PRINT:GOTO 50
248 PO PRINT:POKE32194,C64:PRINT:
   POKE32195,C64:PRINT:GOTO 50
250 PO PRINT:POKE32196,C64:PRINT:
   POKE32197,C64:PRINT:GOTO 50
252 PO PRINT:POKE32198,C64:PRINT:
   POKE32199,C64:PRINT:GOTO 50
254 PO PRINT:POKE32200,C64:PRINT:
   POKE32201,C64:PRINT:GOTO 50
256 PO PRINT:POKE32202,C64:PRINT:
   POKE32203,C64:PRINT:GOTO 50
258 PO PRINT:POKE32204,C64:PRINT:
   POKE32205,C64:PRINT:GOTO 50
260 PO PRINT:POKE32206,C64:PRINT:
   POKE32207,C64:PRINT:GOTO 50
262 PO PRINT:POKE32208,C64:PRINT:
   POKE32209,C64:PRINT:GOTO 50
264 PO PRINT:POKE32210,C64:PRINT:
   POKE32211,C64:PRINT:GOTO 50
266 PO PRINT:POKE32212,C64:PRINT:
   POKE32213,C64:PRINT:GOTO 50
268 PO PRINT:POKE32214,C64:PRINT:
   POKE32215,C64:PRINT:GOTO 50
270 PO PRINT:POKE32216,C64:PRINT:
   POKE32217,C64:PRINT:GOTO 50
272 PO PRINT:POKE32218,C64:PRINT:
   POKE32219,C64:PRINT:GOTO 50
274 PO PRINT:POKE32220,C64:PRINT:
   POKE32221,C64:PRINT:GOTO 50
276 PO PRINT:POKE32222,C64:PRINT:
   POKE32223,C64:PRINT:GOTO 50
278 PO PRINT:POKE32224,C64:PRINT:
   POKE32225,C64:PRINT:GOTO 50
280 PO PRINT:POKE32226,C64:PRINT:
   POKE32227,C64:PRINT:GOTO 50
282 PO PRINT:POKE32228,C64:PRINT:
   POKE32229,C64:PRINT:GOTO 50
284 PO PRINT:POKE32230,C64:PRINT:
   POKE32231,C64:PRINT:GOTO 50
286 PO PRINT:POKE32232,C64:PRINT:
   POKE32233,C64:PRINT:GOTO 50
288 PO PRINT:POKE32234,C64:PRINT:
   POKE32235,C64:PRINT:GOTO 50
290 PO PRINT:POKE32236,C64:PRINT:
   POKE32237,C64:PRINT:GOTO 50
292 PO PRINT:POKE32238,C64:PRINT:
   POKE32239,C64:PRINT:GOTO 50
294 PO PRINT:POKE32240,C64:PRINT:
   POKE32241,C64:PRINT:GOTO 50
296 PO PRINT:POKE32242,C64:PRINT:
   POKE32243,C64:PRINT:GOTO 50
298 PO PRINT:POKE32244,C64:PRINT:
   POKE32245,C64:PRINT:GOTO 50
300 PO PRINT:POKE32246,C64:PRINT:
   POKE32247,C64:PRINT:GOTO 50
302 PO PRINT:POKE32248,C64:PRINT:
   POKE32249,C64:PRINT:GOTO 50
304 PO PRINT:POKE32250,C64:PRINT:
   POKE32251,C64:PRINT:GOTO 50
306 PO PRINT:POKE32252,C64:PRINT:
   POKE32253,C64:PRINT:GOTO 50
308 PO PRINT:POKE32254,C64:PRINT:
   POKE32255,C64:PRINT:GOTO 50
310 PO PRINT:POKE32256,C64:PRINT:
   POKE32257,C64:PRINT:GOTO 50
312 PO PRINT:POKE32258,C64:PRINT:
   POKE32259,C64:PRINT:GOTO 50
314 PO PRINT:POKE32260,C64:PRINT:
   POKE32261,C64:PRINT:GOTO 50
316 PO PRINT:POKE32262,C64:PRINT:
   POKE32263,C64:PRINT:GOTO 50
318 PO PRINT:POKE32264,C64:PRINT:
   POKE32265,C64:PRINT:GOTO 50
320 PO PRINT:POKE32266,C64:PRINT:
   POKE32267,C64:PRINT:GOTO 50
322 PO PRINT:POKE32268,C64:PRINT:
   POKE32269,C64:PRINT:GOTO 50
324 PO PRINT:POKE32270,C64:PRINT:
   POKE32271,C64:PRINT:GOTO 50
326 PO PRINT:POKE32272,C64:PRINT:
   POKE32273,C64:PRINT:GOTO 50
328 PO PRINT:POKE32274,C64:PRINT:
   POKE32275,C64:PRINT:GOTO 50
330 PO PRINT:POKE32276,C64:PRINT:
   POKE32277,C64:PRINT:GOTO 50
332 PO PRINT:POKE32278,C64:PRINT:
   POKE32279,C64:PRINT:GOTO 50
334 PO PRINT:POKE32280,C64:PRINT:
   POKE32281,C64:PRINT:GOTO 50
336 PO PRINT:POKE32282,C64:PRINT:
   POKE32283,C64:PRINT:GOTO 50
338 PO PRINT:POKE32284,C64:PRINT:
   POKE32285,C64:PRINT:GOTO 50
340 PO PRINT:POKE32286,C64:PRINT:
   POKE32287,C64:PRINT:GOTO 50
342 PO PRINT:POKE32288,C64:PRINT:
   POKE32289,C64:PRINT:GOTO 50
344 PO PRINT:POKE32290,C64:PRINT:
   POKE32291,C64:PRINT:GOTO 50
346 PO PRINT:POKE32292,C64:PRINT:
   POKE32293,C64:PRINT:GOTO 50
348 PO PRINT:POKE32294,C64:PRINT:
   POKE32295,C64:PRINT:GOTO 50
350 PO PRINT:POKE32296,C64:PRINT:
   POKE32297,C64:PRINT:GOTO 50
352 PO PRINT:POKE32298,C64:PRINT:
   POKE32299,C64:PRINT:GOTO 50
354 PO PRINT:POKE32300,C64:PRINT:
   POKE32301,C64:PRINT:GOTO 50
356 PO PRINT:POKE32302,C64:PRINT:
   POKE32303,C64:PRINT:GOTO 50
358 PO PRINT:POKE32304,C64:PRINT:
   POKE32305,C64:PRINT:GOTO 50
360 PO PRINT:POKE32306,C64:PRINT:
   POKE32307,C64:PRINT:GOTO 50
362 PO PRINT:POKE32308,C64:PRINT:
   POKE32309,C64:PRINT:GOTO 50
364 PO PRINT:POKE32310,C64:PRINT:
   POKE32311,C64:PRINT:GOTO 50
366 PO PRINT:POKE32312,C64:PRINT:
   POKE32313,C64:PRINT:GOTO 50
368 PO PRINT:POKE32314,C64:PRINT:
   POKE32315,C64:PRINT:GOTO 50
370 PO PRINT:POKE32316,C64:PRINT:
   POKE32317,C64:PRINT:GOTO 50
372 PO PRINT:POKE32318,C64:PRINT:
   POKE32319,C64:PRINT:GOTO 50
374 PO PRINT:POKE32320,C64:PRINT:
   POKE32321,C64:PRINT:GOTO 50
376 PO PRINT:POKE32322,C64:PRINT:
   POKE32323,C64:PRINT:GOTO 50
378 PO PRINT:POKE32324,C64:PRINT:
   POKE32325,C64:PRINT:GOTO 50
380 PO PRINT:POKE32326,C64:PRINT:
   POKE32327,C64:PRINT:GOTO 50
382 PO PRINT:POKE32328,C64:PRINT:
   POKE32329,C64:PRINT:GOTO 50
384 PO PRINT:POKE32330,C64:PRINT:
   POKE32331,C64:PRINT:GOTO 50
386 PO PRINT:POKE32332,C64:PRINT:
   POKE32333,C64:PRINT:GOTO 50
388 PO PRINT:POKE32334,C64:PRINT:
   POKE32335,C64:PRINT:GOTO 50
390 PO PRINT:POKE32336,C64:PRINT:
   POKE32337,C64:PRINT:GOTO 50
392 PO PRINT:POKE32338,C64:PRINT:
   POKE32339,C64:PRINT:GOTO 50
394 PO PRINT:POKE32340,C64:PRINT:
   POKE32341,C64:PRINT:GOTO 50
396 PO PRINT:POKE32342,C64:PRINT:
   POKE32343,C64:PRINT:GOTO 50
398 PO PRINT:POKE32344,C64:PRINT:
   POKE32345,C64:PRINT:GOTO 50
400 PO PRINT:POKE32346,C64:PRINT:
   POKE32347,C64:PRINT:GOTO 50
402 PO PRINT:POKE32348,C64:PRINT:
   POKE32349,C64:PRINT:GOTO 50
404 PO PRINT:POKE32350,C64:PRINT:
   POKE32351,C64:PRINT:GOTO 50
406 PO PRINT:POKE32352,C64:PRINT:
   POKE32353,C64:PRINT:GOTO 50
408 PO PRINT:POKE32354,C64:PRINT:
   POKE32355,C64:PRINT:GOTO 50
410 PO PRINT:POKE32356,C64:PRINT:
   POKE32357,C64:PRINT:GOTO 50
412 PO PRINT:POKE32358,C64:PRINT:
   POKE32359,C64:PRINT:GOTO 50
414 PO PRINT:POKE32360,C64:PRINT:
   POKE32361,C64:PRINT:GOTO 50
416 PO PRINT:POKE32362,C64:PRINT:
   POKE32363,C64:PRINT:GOTO 50
418 PO PRINT:POKE32364,C64:PRINT:
   POKE32365,C64:PRINT:GOTO 50
420 PO PRINT:POKE32366,C64:PRINT:
   POKE32367,C64:PRINT:GOTO 50
422 PO PRINT:POKE32368,C64:PRINT:
   POKE32369,C64:PRINT:GOTO 50
424 PO PRINT:POKE32370,C64:PRINT:
   POKE32371,C64:PRINT:GOTO 50
426 PO PRINT:POKE32372,C64:PRINT:
   POKE32373,C64:PRINT:GOTO 50
428 PO PRINT:POKE32374,C64:PRINT:
   POKE32375,C64:PRINT:GOTO 50
430 PO PRINT:POKE32376,C64:PRINT:
   POKE32377,C64:PRINT:GOTO 50
432 PO PRINT:POKE32378,C64:PRINT:
   POKE32379,C64:PRINT:GOTO 50
434 PO PRINT:POKE32380,C64:PRINT:
   POKE32381,C64:PRINT:GOTO 50
436 PO PRINT:POKE32382,C64:PRINT:
   POKE32383,C64:PRINT:GOTO 50
438 PO PRINT:POKE32384,C64:PRINT:
   POKE32385,C64:PRINT:GOTO 50
440 PO PRINT:POKE32386,C64:PRINT:
   POKE32387,C64:PRINT:GOTO 50
442 PO PRINT:POKE32388,C64:PRINT:
   POKE32389,C64:PRINT:GOTO 50
444 PO PRINT:POKE32390,C64:PRINT:
   POKE32391,C64:PRINT:GOTO 50
446 PO PRINT:POKE32392,C64:PRINT:
   POKE32393,C64:PRINT:GOTO 50
448 PO PRINT:POKE32394,C64:PRINT:
   POKE32395,C64:PRINT:GOTO 50
450 PO PRINT:POKE32396,C64:PRINT:
   POKE32397,C64:PRINT:GOTO 50
452 PO PRINT:POKE32398,C64:PRINT:
   POKE32399,C64:PRINT:GOTO 50
454 PO PRINT:POKE32400,C64:PRINT:
   POKE32401,C64:PRINT:GOTO 50
456 PO PRINT:POKE32402,C64:PRINT:
   POKE32403,C64:PRINT:GOTO 50
458 PO PRINT:POKE32404,C64:PRINT:
   POKE32405,C64:PRINT:GOTO 50
460 PO PRINT:POKE32406,C64:PRINT:
   POKE32407,C64:PRINT:GOTO 50
462 PO PRINT:POKE32408,C64:PRINT:
   POKE32409,C64:PRINT:GOTO 50
464 PO PRINT:POKE32410,C64:PRINT:
   POKE32411,C64:PRINT:GOTO 50
466 PO PRINT:POKE32412,C64:PRINT:
   POKE32413,C64:PRINT:GOTO 50
468 PO PRINT:POKE32414,C64:PRINT:
   POKE32415,C64:PRINT:GOTO 50
470 PO PRINT:POKE32416,C64:PRINT:
   POKE32417,C64:PRINT:GOTO 50
472 PO PRINT:POKE32418,C64:PRINT:
   POKE32419,C64:PRINT:GOTO 50
474 PO PRINT:POKE32420,C64:PRINT:
   POKE32421,C64:PRINT:GOTO 50
476 PO PRINT:POKE32422,C64:PRINT:
   POKE32423,C64:PRINT:GOTO 50
478 PO PRINT:POKE32424,C64:PRINT:
   POKE32425,C64:PRINT:GOTO 50
480 PO PRINT:POKE32426,C64:PRINT:
   POKE32427,C64:PRINT:GOTO 50
482 PO PRINT:POKE32428,C64:PRINT:
   POKE32429,C64:PRINT:GOTO 50
484 PO PRINT:POKE32430,C64:PRINT:
   POKE32431,C64:PRINT:GOTO 50
486 PO PRINT:POKE32432,C64:PRINT:
   POKE32433,C64:PRINT:GOTO 50
488 PO PRINT:POKE32434,C64:PRINT:
   POKE32435,C64:PRINT:GOTO 50
490 PO PRINT:POKE32436,C64:PRINT:
   POKE32437,C64:PRINT:GOTO 50
492 PO PRINT:POKE32438,C64:PRINT:
   POKE32439,C64:PRINT:GOTO 50
494 PO PRINT:POKE32440,C64:PRINT:
   POKE32441,C64:PRINT:GOTO 50
496 PO PRINT:POKE32442,C64:PRINT:
   POKE32443,C64:PRINT:GOTO 50
498 PO PRINT:POKE32444,C64:PRINT:
   POKE32445,C64:PRINT:GOTO 50
500 PO PRINT:POKE32446,C64:PRINT:
   POKE32447,C64:PRINT:GOTO 50
502 PO PRINT:POKE32448,C64:PRINT:
   POKE32449,C64:PRINT:GOTO 50
504 PO PRINT:POKE32450,C64:PRINT:
   POKE32451,C64:PRINT:GOTO 50
506 PO PRINT:POKE32452,C64:PRINT:
   POKE32453,C64:PRINT:GOTO 50
508 PO PRINT:POKE32454,C64:PRINT:
   POKE32455,C64:PRINT:GOTO 50
510 PO PRINT:POKE32456,C64:PRINT:
   POKE32457,C64:PRINT:GOTO 50
512 PO PRINT:POKE32458,C64:PRINT:
   POKE32459,C64:PRINT:GOTO 50
514 PO PRINT:POKE32460,C64:PRINT:
   POKE32461,C64:PRINT:GOTO 50
516 PO PRINT:POKE32462,C64:PRINT:
   POKE32463,C64:PRINT:GOTO 50
518 PO PRINT:POKE32464,C64:PRINT:
   POKE32465,C64:PRINT:GOTO 50
520 PO PRINT:POKE32466,C64:PRINT:
   POKE32467,C64:PRINT:GOTO 50
522 PO PRINT:POKE32468,C64:PRINT:
   POKE32469,C64:PRINT:GOTO 50
524 PO PRINT:POKE32470,C64:PRINT:
   POKE32471,C64:PRINT:GOTO 50
526 PO PRINT:POKE32472,C64:PRINT:
   POKE32473,C64:PRINT:GOTO 50
528 PO PRINT:POKE32474,C64:PRINT:
   POKE32475,C64:PRINT:GOTO 50
530 PO PRINT:POKE32476,C64:PRINT:
   POKE32477,C64:PRINT:GOTO 50
532 PO PRINT:POKE32478,C64:PRINT:
   POKE32479,C64:PRINT:GOTO 50
534 PO PRINT:POKE32480,C64:PRINT:
   POKE32481,C64:PRINT:GOTO 50
536 PO PRINT:POKE32482,C64:PRINT:
   POKE32483,C64:PRINT:GOTO 50
538 PO PRINT:POKE32484,C64:PRINT:
   POKE32485,C64:PRINT:GOTO 50
540 PO PRINT:POKE32486,C64:PRINT:
   POKE32487,C64:PRINT:GOTO 50
542 PO PRINT:POKE32488,C64:PRINT:
   POKE32489,C64:PRINT:GOTO 50
544 PO PRINT:POKE32490,C64:PRINT:
   POKE32491,C64:PRINT:GOTO 50
546 PO PRINT:POKE32492,C64:PRINT:
   POKE32493,C64:PRINT:GOTO 50
548 PO PRINT:POKE32494,C64:PRINT:
   POKE32495,C64:PRINT:GOTO 50
550 PO PRINT:POKE32496,C64:PRINT:
   POKE32497,C64:PRINT:GOTO 50
552 PO PRINT:POKE32498,C64:PRINT:
   POKE32499,C64:PRINT:GOTO 50
554 PO PRINT:POKE32500,C64:PRINT:
   POKE32501,C64:PRINT:GOTO 50
556 PO PRINT:POKE32502,C64:PRINT:
   POKE32503,C64:PRINT:GOTO 50
558 PO PRINT:POKE32504,C64:PRINT:
   POKE32505,C64:PRINT:GOTO 50
560 PO PRINT:POKE32506,C64:PRINT:
   POKE32507,C64:PRINT:GOTO 50
562 PO PRINT:POKE32508,C64:PRINT:
   POKE32509,C64:PRINT:GOTO 50
564 PO PRINT:POKE32510,C64:PRINT:
   POKE32511,C64:PRINT:GOTO 50
566 PO PRINT:POKE32512,C64:PRINT:
   POKE32513,C64:PRINT:GOTO 50
568 PO PRINT:POKE32514,C64:PRINT:
   POKE32515,C64:PRINT:GOTO 50
570 PO PRINT:POKE32516,C64:PRINT:
   POKE32517,C64:PRINT:GOTO 50
572 PO PRINT:POKE32518,C64:PRINT:
   POKE32519,C64:PRINT:GOTO 50
574 PO PRINT:POKE32520,C64:PRINT:
   POKE32521,C64:PRINT:GOTO 50
576 PO PRINT:POKE32522,C64:PRINT:
   POKE32523,C64:PRINT:GOTO 50
578 PO PRINT:POKE32524,C64:PRINT:
   POKE32525,C64:PRINT:GOTO 50
580 PO PRINT:POKE32526,C64:PRINT:
   POKE32527,C64:PRINT:GOTO 50
582 PO PRINT:POKE32528,C64:PRINT:
   POKE32529,C64:PRINT:GOTO 50
584 PO PRINT:POKE32530,C64:PRINT:
   POKE32531,C64:PRINT:GOTO 50
586 PO PRINT:POKE32532,C64:PRINT:
   POKE32533,C64:PRINT:GOTO 50
588 PO PRINT:POKE32534,C64:PRINT:
   POKE32535,C64:PRINT:GOTO 50
590 PO PRINT:POKE32536,C64:PRINT:
   POKE32537,C64:PRINT:GOTO 50
592 PO PRINT:POKE32538,C64:PRINT:
   POKE32539,C64:PRINT:GOTO 50
594 PO PRINT:POKE32540,C64:PRINT:
   POKE32541,C64:PRINT:GOTO 50
596 PO PRINT:POKE32542,C64:PRINT:
   POKE32543,C64:PRINT:GOTO 50
598 PO PRINT:POKE32544,C64:PRINT:
   POKE32545,C64:PRINT:GOTO 50
600 PO PRINT:POKE32546,C64:PRINT:
   POKE32547,C64:PRINT:GOTO 50
602 PO PRINT:POKE32548,C64:PRINT:
   POKE32549,C64:PRINT:GOTO 50
604 PO PRINT:POKE32550,C64:PRINT:
   POKE32551,C64:PRINT:GOTO 50
606 PO PRINT:POKE32552,C64:PRINT:
   POKE32553,C64:PRINT:GOTO 50
608 PO PRINT:POKE32
```

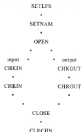
1

# Machine Code Disk Programming

If you can handle your disk drive in Basic, Machine Code programs using *comms* very naturally — demands slightly more work, but the *Kernal* and *DOS* still do nearly all the work for you. Machine code disk commands have a clear fail, with Basic — they both use the *Kernal*, Commodore's I/O routines. The difference is that machine code routines, especially the handling, happen at lightning speed. This article will concentrate on the handling, the routines, and practical examples.

## The Record

By the way, that's Commodore's spelling, not mine! Note that all the below routines are called with the ISR instructions, with the appropriate carrying data. Below is a flowchart for the use of the routines.



*Continuing his series on disk drives, Fergal Moore untouches the mysteries of Machine Code*

There is a summary of the necessary conditions.

STUDIES

A FILE NUMBER  
X INDEX NUMBER  
Y SECONDARY ADDRESS

This sets up parameters for use with any disk I/O, and is equivalent to the first three numbers in an OPEN statement.

**BEST COPY AVAILABLE**

```
A NAME LENGTH
A LOW BYTE OF START OF
NAME
T HIGH BYTE OF START OF
NAME
```

Seize a name for disk I/O. Note that for disks, a name must always be specified, except opening a file channel. (5.6.10.15.1.15)

[illegible]

### FROM THE EDITORIAL BOARD

Use to open a file after SETLFS and SETMAID

**Abstract**

## FILE SHARING

☐ Save the specified file

## CHILDREN

**THE**

Set up a channel for input, after  
using the OPEN command

## CHANGING LEADERS

## THE FUTURE OF THE FIRM

Set up a channel for output, adjust  
using the CPU% command.

**Citation:**

## NOTES

```
inputs data from the input channel
defined by CHIN, storing it in the
accumulator. Equivalent to GET
```

## CHROMATOPHORE

## A DATA COMMENT

Outputs the data in the accumulation to the output channel defined in `CHOUT`. Equivalent to `PRINT`.

**CLARENCE BERRY**

### THE FARMALABITERS' BROTHERHOOD

Return all input to the keyboard, and output to the screen. Use after finishing your code. ID:

## Example

What are two assembly layouts to demonstrate the use of the above revision?



[illegible][illegible]

For more information on KERNAL routines, see the Programmer's Reference Guide. Next time, I will present a complete DOS which demonstrates the above routines in a practical situation.

## 000000

Unfortunately, we left four things out from our June's installment of *Mac OS Code Book Programming*, so we're reviewing them below:

[illegible][illegible][illegible]

```

00 ENTRY *****
*****
00 ENTRY
      "
00 ENTRY          UPSTREAM FILE
      "
00 ENTRY
      "
00 ENTRY *****
*****
00 PRINT TRANSACTION FILE PRINT
*****
70 PRINT "ENTER SEARCHED FILE NAME"
00 PRINT ENTER END TO EXIT
00 IF 1=0 THEN END ELSE TRAN FILE
    10 READ DEL VAR
    110 *****
*****
100 REW FID OUT WITH SKIP OF 70
END

```

[illegible]

LEAD: 1000000

[illegible]

PROTECT FILE

[illegible]

```

04  *****
05  *** SELECTED CASES OUTSIDE
06  E. PROVINCE. (INFORM) ***
07  *** EXPERIMENTAL. SPECIAL
08  PROGRAMS IN CASES, RESULTS, &
09  THE INFORMED ***
10  *** (CASE) CASE, SOURCE: I
11  CASE ***
12  *** EXPERIMENTAL, DISCLOSED,
13  INSTANT WITHIN FREEDOM OF
14  ACT ***
15  *** EXPERIMENTAL. ALICE CO
16  EXPERIMENTAL DISCLOSED, SOURCE,
17  DISCLOSED ***
18  *** (CASE) CASE, TURN OUT
19  ***
20  *** EXPERIMENTAL. SOURCE
21  *** EXPERIMENTAL. CONTROLLING AND
22  SOURCE (INFORMED) ***
23  *** EXPERIMENTAL. ALICE CO
24  EXPERIMENTAL. DISCLOSED, SOURCE
25  ***
26  *** (CASE) CASE, SOURCE: I
27  CASE ***
28  *** EXPERIMENTAL. DISCLOSED,
29  INSTANT WITHIN FREEDOM OF
30  ACT ***
31  *****

```

# Inside the

*Continuing his series on programming the 1541 drive, Fergal Moane offers some insights into the internals of your machine*

*By Fergal Moane*

Now that we've gained a fair knowledge of machine code and disk editing, it's time to turn our attention to the internal memory of the 1541. Inside the 1541 lies a dedicated computer. It has a 6502 processor (as in the 64), 16K ROM, 16K RAM, and interface chips.

Commodore's designers have for once had foresight, and given us a host of advanced commands to manipulate the drive's internal memory. You can use the drive's ROM routines, or write your own in the RAM buffers. This allows advanced protection routines and disk caches to be created. I've even seen a program which runs the drive's 6502 to calculate numbers for sector graphics routines, effectively using the drive as a rejigged alarm processor, speeding up things no end.

## The Commands

The three commands below are the approximate equivalent of PEEK, POKE and RTS. Using these commands, it's possible to create your own machine code, or use the DOS routines. Unfortunately, Commodore has never published disassemblies of its disk ROM, so use of it is extremely limited. The best way to use the disk memory is to lay a machine code monitor which allows assembly and disassembly to disk memory (GATEL and TRILONIC have cartridges with monitors possessing this facility).

### MEMORY-READ

```
PRINT #15, "M-R" CHR$(0)
CHR$(0) CHR$(number)
CHR$(0) is the low byte of the
```

address in DOS that is to be read. CHR\$(0) is the high byte of the address.

CHR\$(number) is an optional parameter which indicates the number of bytes to be read.

The percentage (as OPEN 15.8,15 has been performed earlier in the program (as it always should be)) A GET #15 will read the byte from the error channel, performing a PEEK of the address specified by the CHR\$(0) codes.

### MEMORY-WRITE

```
PRINT #15, "M-W" CHR$(0)
CHR$(0) CHR$(number) CHR$(
data) CHR$(data) etc .
```

Number obviously specifies the number of bytes to be written. The number of points of data should correspond to the number in the third CHR\$(number). The maximum amount of data that can be sent at one time is 34 bytes.

### MEMORY-EXECUTE

```
PRINT #15, "M-E" CHR$(0)
CHR$(0)
```

This command will call and execute a machine code program that resides in the DOS memory. This routine should end with a RTS instruction. DOS routines can be used with this command.

### USER COMMANDS

```
PRINT #15, "U"
```

See the table below for possible values of N. These commands allow a jump table to be set up, as there is

enough room for a JMP instruction to your routine. This means that by using registers, you can access a table of routines quickly and easily, even from Basic. All the below User commands access buffer 2, the unused buffer capable for machine code.

U0 or UC	jump to \$0500
U4 or UD	jump to \$0504
U5 or UE	jump to \$0505
U6 or UF	jump to \$0506
U7 or UG	jump to \$050C
U8 or UH	jump to \$050F
U9 or UI	jump to \$FFFF
U. or UJ	power-up vector, runs drive

## Examples

Here are some simple examples which demonstrate the above theory. They may be useful to disk utility programs.

**EXAMPLE 1** - open on the current disk.

```
10 OPEN 15,8,15,"T"
20 PRINT #15, "M-R" CHR$(255)
CHR$(0)
30 GET #15, X5: X5 = X5 + CHR$(0)
40 PRINT #15, "M-R" CHR$(255)
CHR$(0)
50 GET #15, Y5: Y5 = Y5 + CHR$(0)
60 A=ASC(X5)+256*ASC(Y5)
70 PRINTA:PRINT "KILobytes"
80 PRINTA/4 "KILobytes"
FREE:PRINTA/4 "KILobytes"
90 CLOSE 15
```

This reads the blocks free on a disk, and calculates the number of kilobytes available by dividing by four. X5 and

# e 1541

Y1 contains the low and high bytes of the blocks free.

## EXAMPLE 1—alignment adjuster and woodpecker remover

```
10 INPUT "WOODPECKER (Y/N)? "
10 JWS
20 SPWS < > "Y" THEN B=B+128
30 INPUT "HALF TRACK SEEKER (Y/N)? AS
40 IF AS < > "Y" THEN B=B+64
50 INPUT "LOADING ATTEMPTS (0-9)? J.
60 B=B-L. OPEN 15, 8, 15
70 PRINT # 15, "M-W" CHR(140)
CHR(5) CHR(1) CHR(8)
80 CLOSE 15
```

This program alters the vital location 168 in disk RAM. It controls the number of read attempts to be made, i.e. how many times the disk drive will try to read the sector. This is usually fine. The "woodpecker" is made when the disk head bounces off the end stop and is avoided. Also, if your disk drive is out of alignment, the half track reader will increase the chance of loading a misaligned program.

## EXAMPLE 2—disk name

```
10 OPEN 15, 8, 15, "P"
20 PRINT # 15, "M-R" CHR(144)
CHR(1) CHR(8) (140)
30 INPUT # 15, NAME$
```

## 40 PRINT NAME\$; CLOSE 15

This reads the entire letter name of the random disk. Note the CHR(140) to define the number of locations to be read and the use of INPUT # as each multiple characters.

When fiddling around with your disk drive, it makes sense to tape the disk out of the drive. This means that if a dull value hits a sensitive spot, your disk will not suffer the consequences. Remember that you cannot damage the drive by software, and switching it off and on again will remove everything to normal.

## Memory Map

There's little point in standing around with drive memory if you don't know what you're doing. Below is a 1541 memory map which details some of the more interesting features. I came across it one day while providing DOS documentation, so you'll need a good disk monitor to experiment with DOS routines.

RAM 0000-0FFF 0000-2047  
ROM 0000-FFFF 2048-65535

HEX	DECIMAL	DESCRIPTION	HEX	DECIMAL	DESCRIPTION
0000	0	COMMAND CODE FOR BUFFER 1	0000-0003	00-03	BUFFER POINTER
0001	1	COMMAND CODE FOR BUFFER 1	0000-0004	00-04	ADDRESS OF BUFFER 1
0002	2	COMMAND CODE FOR BUFFER 2	0000-0005	00-05	ADDRESS OF BUFFER 1
0003	3	COMMAND CODE FOR BUFFER 2	0000-0006	00-06	ADDRESS OF BUFFER 1
0004	4	COMMAND CODE FOR BUFFER 3	0000-0007	00-07	ADDRESS OF BUFFER 1
0005-0007	5-7	TRACK AND SECTOR FOR BUFFER 1	0000-0008	00-08	ADDRESS OF BUFFER 2
0008-0009	8-9	TRACK AND SECTOR FOR BUFFER 1	0000-0009	00-09	ADDRESS OF BUFFER 2
000A-000B	10-11	TRACK AND SECTOR FOR BUFFER 2	0000-000A	00-10	ADDRESS OF BUFFER 2
000C-000D	12-13	TRACK AND SECTOR FOR BUFFER 2	0000-000B	00-11	ADDRESS OF BUFFER 2
000E-000F	14-15	TRACK AND SECTOR FOR BUFFER 2	0000-000C	00-12	ADDRESS OF BUFFER 2
0010-0011	16-17	SECTOR NUMBER	0000-000D	00-13	ADDRESS OF BUFFER 2
0012-0013	18-19	SECTOR NUMBER	0000-000E	00-14	ADDRESS OF BUFFER 2
0014-0015	20-21	SECTOR NUMBER	0000-000F	00-15	ADDRESS OF BUFFER 2
0016-0017	22-23	SECTOR NUMBER	0000-0010	00-16	ADDRESS OF BUFFER 2
0018-0019	24-25	SECTOR NUMBER	0000-0011	00-17	ADDRESS OF BUFFER 2
001A-001B	26-27	SECTOR NUMBER	0000-0012	00-18	ADDRESS OF BUFFER 2
001C-001D	28-29	SECTOR NUMBER	0000-0013	00-19	ADDRESS OF BUFFER 2
001E-001F	30-31	SECTOR NUMBER	0000-0014	00-20	ADDRESS OF BUFFER 2
0020-0021	32-33	SECTOR NUMBER	0000-0015	00-21	ADDRESS OF BUFFER 2
0022-0023	34-35	SECTOR NUMBER	0000-0016	00-22	ADDRESS OF BUFFER 2
0024-0025	36-37	SECTOR NUMBER	0000-0017	00-23	ADDRESS OF BUFFER 2
0026-0027	38-39	SECTOR NUMBER	0000-0018	00-24	ADDRESS OF BUFFER 2
0028-0029	40-41	SECTOR NUMBER	0000-0019	00-25	ADDRESS OF BUFFER 2
002A-002B	42-43	SECTOR NUMBER	0000-001A	00-26	ADDRESS OF BUFFER 2
002C-002D	44-45	SECTOR NUMBER	0000-001B	00-27	ADDRESS OF BUFFER 2
002E-002F	46-47	SECTOR NUMBER	0000-001C	00-28	ADDRESS OF BUFFER 2
0030-0031	48-49	SECTOR NUMBER	0000-001D	00-29	ADDRESS OF BUFFER 2
0032-0033	50-51	SECTOR NUMBER	0000-001E	00-30	ADDRESS OF BUFFER 2
0034-0035	52-53	SECTOR NUMBER	0000-001F	00-31	ADDRESS OF BUFFER 2
0036-0037	54-55	SECTOR NUMBER	0000-0020	00-32	ADDRESS OF BUFFER 2
0038-0039	56-57	SECTOR NUMBER	0000-0021	00-33	ADDRESS OF BUFFER 2
003A-003B	58-59	SECTOR NUMBER	0000-0022	00-34	ADDRESS OF BUFFER 2
003C-003D	60-61	SECTOR NUMBER	0000-0023	00-35	ADDRESS OF BUFFER 2
003E-003F	62-63	SECTOR NUMBER	0000-0024	00-36	ADDRESS OF BUFFER 2
0040-0041	64-65	SECTOR NUMBER	0000-0025	00-37	ADDRESS OF BUFFER 2
0042-0043	66-67	SECTOR NUMBER	0000-0026	00-38	ADDRESS OF BUFFER 2
0044-0045	68-69	SECTOR NUMBER	0000-0027	00-39	ADDRESS OF BUFFER 2
0046-0047	70-71	SECTOR NUMBER	0000-0028	00-40	ADDRESS OF BUFFER 2
0048-0049	72-73	SECTOR NUMBER	0000-0029	00-41	ADDRESS OF BUFFER 2
004A-004B	74-75	SECTOR NUMBER	0000-002A	00-42	ADDRESS OF BUFFER 2
004C-004D	76-77	SECTOR NUMBER	0000-002B	00-43	ADDRESS OF BUFFER 2
004E-004F	78-79	SECTOR NUMBER	0000-002C	00-44	ADDRESS OF BUFFER 2
0050-0051	80-81	SECTOR NUMBER	0000-002D	00-45	ADDRESS OF BUFFER 2
0052-0053	82-83	SECTOR NUMBER	0000-002E	00-46	ADDRESS OF BUFFER 2
0054-0055	84-85	SECTOR NUMBER	0000-002F	00-47	ADDRESS OF BUFFER 2
0056-0057	86-87	SECTOR NUMBER	0000-0030	00-48	ADDRESS OF BUFFER 2
0058-0059	88-89	SECTOR NUMBER	0000-0031	00-49	ADDRESS OF BUFFER 2
005A-005B	90-91	SECTOR NUMBER	0000-0032	00-50	ADDRESS OF BUFFER 2
005C-005D	92-93	SECTOR NUMBER	0000-0033	00-51	ADDRESS OF BUFFER 2
005E-005F	94-95	SECTOR NUMBER	0000-0034	00-52	ADDRESS OF BUFFER 2
0060-0061	96-97	SECTOR NUMBER	0000-0035	00-53	ADDRESS OF BUFFER 2
0062-0063	98-99	SECTOR NUMBER	0000-0036	00-54	ADDRESS OF BUFFER 2
0064-0065	100-101	SECTOR NUMBER	0000-0037	00-55	ADDRESS OF BUFFER 2
0066-0067	102-103	SECTOR NUMBER	0000-0038	00-56	ADDRESS OF BUFFER 2
0068-0069	104-105	SECTOR NUMBER	0000-0039	00-57	ADDRESS OF BUFFER 2
006A-006B	106-107	SECTOR NUMBER	0000-003A	00-58	ADDRESS OF BUFFER 2
006C-006D	108-109	SECTOR NUMBER	0000-003B	00-59	ADDRESS OF BUFFER 2
006E-006F	110-111	SECTOR NUMBER	0000-003C	00-60	ADDRESS OF BUFFER 2
0070-0071	112-113	SECTOR NUMBER	0000-003D	00-61	ADDRESS OF BUFFER 2
0072-0073	114-115	SECTOR NUMBER	0000-003E	00-62	ADDRESS OF BUFFER 2
0074-0075	116-117	SECTOR NUMBER	0000-003F	00-63	ADDRESS OF BUFFER 2
0076-0077	118-119	SECTOR NUMBER	0000-0040	00-64	ADDRESS OF BUFFER 2
0078-0079	120-121	SECTOR NUMBER	0000-0041	00-65	ADDRESS OF BUFFER 2
007A-007B	122-123	SECTOR NUMBER	0000-0042	00-66	ADDRESS OF BUFFER 2
007C-007D	124-125	SECTOR NUMBER	0000-0043	00-67	ADDRESS OF BUFFER 2
007E-007F	126-127	SECTOR NUMBER	0000-0044	00-68	ADDRESS OF BUFFER 2
0080-0081	128-129	SECTOR NUMBER	0000-0045	00-69	ADDRESS OF BUFFER 2
0082-0083	130-131	SECTOR NUMBER	0000-0046	00-70	ADDRESS OF BUFFER 2
0084-0085	132-133	SECTOR NUMBER	0000-0047	00-71	ADDRESS OF BUFFER 2
0086-0087	134-135	SECTOR NUMBER	0000-0048	00-72	ADDRESS OF BUFFER 2
0088-0089	136-137	SECTOR NUMBER	0000-0049	00-73	ADDRESS OF BUFFER 2
008A-008B	138-139	SECTOR NUMBER	0000-004A	00-74	ADDRESS OF BUFFER 2
008C-008D	140-141	SECTOR NUMBER	0000-004B	00-75	ADDRESS OF BUFFER 2
008E-008F	142-143	SECTOR NUMBER	0000-004C	00-76	ADDRESS OF BUFFER 2
0090-0091	144-145	SECTOR NUMBER	0000-004D	00-77	ADDRESS OF BUFFER 2
0092-0093	146-147	SECTOR NUMBER	0000-004E	00-78	ADDRESS OF BUFFER 2
0094-0095	148-149	SECTOR NUMBER	0000-004F	00-79	ADDRESS OF BUFFER 2
0096-0097	150-151	SECTOR NUMBER	0000-0050	00-80	ADDRESS OF BUFFER 2
0098-0099	152-153	SECTOR NUMBER	0000-0051	00-81	ADDRESS OF BUFFER 2
009A-009B	154-155	SECTOR NUMBER	0000-0052	00-82	ADDRESS OF BUFFER 2
009C-009D	156-157	SECTOR NUMBER	0000-0053	00-83	ADDRESS OF BUFFER 2
009E-009F	158-159	SECTOR NUMBER	0000-0054	00-84	ADDRESS OF BUFFER 2
00A0-00A1	160-161	SECTOR NUMBER	0000-0055	00-85	ADDRESS OF BUFFER 2
00A2-00A3	162-163	SECTOR NUMBER	0000-0056	00-86	ADDRESS OF BUFFER 2
00A4-00A5	164-165	SECTOR NUMBER	0000-0057	00-87	ADDRESS OF BUFFER 2
00A6-00A7	166-167	SECTOR NUMBER	0000-0058	00-88	ADDRESS OF BUFFER 2
00A8-00A9	168-169	SECTOR NUMBER	0000-0059	00-89	ADDRESS OF BUFFER 2
00AA-00AB	170-171	SECTOR NUMBER	0000-005A	00-90	ADDRESS OF BUFFER 2
00AC-00AD	172-173	SECTOR NUMBER	0000-005B	00-91	ADDRESS OF BUFFER 2
00AE-00AF	174-175	SECTOR NUMBER	0000-005C	00-92	ADDRESS OF BUFFER 2
00B0-00B1	176-177	SECTOR NUMBER	0000-005D	00-93	ADDRESS OF BUFFER 2
00B2-00B3	178-179	SECTOR NUMBER	0000-005E	00-94	ADDRESS OF BUFFER 2
00B4-00B5	180-181	SECTOR NUMBER	0000-005F	00-95	ADDRESS OF BUFFER 2
00B6-00B7	182-183	SECTOR NUMBER	0000-0060	00-96	ADDRESS OF BUFFER 2
00B8-00B9	184-185	SECTOR NUMBER	0000-0061	00-97	ADDRESS OF BUFFER 2
00BA-00BB	186-187	SECTOR NUMBER	0000-0062	00-98	ADDRESS OF BUFFER 2
00BC-00BD	188-189	SECTOR NUMBER	0000-0063	00-99	ADDRESS OF BUFFER 2
00BE-00BF	190-191	SECTOR NUMBER	0000-0064	00-100	ADDRESS OF BUFFER 2
00C0-00C1	192-193	SECTOR NUMBER	0000-0065	00-101	ADDRESS OF BUFFER 2
00C2-00C3	194-195	SECTOR NUMBER	0000-0066	00-102	ADDRESS OF BUFFER 2
00C4-00C5	196-197	SECTOR NUMBER	0000-0067	00-103	ADDRESS OF BUFFER 2
00C6-00C7	198-199	SECTOR NUMBER	0000-0068	00-104	ADDRESS OF BUFFER 2
00C8-00C9	200-201	SECTOR NUMBER	0000-0069	00-105	ADDRESS OF BUFFER 2
00CA-00CB	202-203	SECTOR NUMBER	0000-006A	00-106	ADDRESS OF BUFFER 2
00CC-00CD	204-205	SECTOR NUMBER	0000-006B	00-107	ADDRESS OF BUFFER 2
00CE-00CF	206-207	SECTOR NUMBER	0000-006C	00-108	ADDRESS OF BUFFER 2
00D0-00D1	208-209	SECTOR NUMBER	0000-006D	00-109	ADDRESS OF BUFFER 2
00D2-00D3	210-211	SECTOR NUMBER	0000-006E	00-110	ADDRESS OF BUFFER 2
00D4-00D5	212-213	SECTOR NUMBER	0000-006F	00-111	ADDRESS OF BUFFER 2
00D6-00D7	214-215	SECTOR NUMBER	0000-0070	00-112	ADDRESS OF BUFFER 2
00D8-00D9	216-217	SECTOR NUMBER	0000-0071	00-113	ADDRESS OF BUFFER 2
00DA-00DB	218-219	SECTOR NUMBER	0000-0072	00-114	ADDRESS OF BUFFER 2
00DC-00DD	220-221	SECTOR NUMBER	0000-0073	00-115	ADDRESS OF BUFFER 2
00DE-00DF	222-223	SECTOR NUMBER	0000-0074	00-116	ADDRESS OF BUFFER 2
00E0-00E1	224-225	SECTOR NUMBER	0000-0075	00-117	ADDRESS OF BUFFER 2
00E2-00E3	226-227	SECTOR NUMBER	0000-0076	00-118	ADDRESS OF BUFFER 2
00E4-00E5	228-229	SECTOR NUMBER	0000-0077	00-119	ADDRESS OF BUFFER 2
00E6-00E7	230-231	SECTOR NUMBER	0000-0078	00-120	ADDRESS OF BUFFER 2
00E8-00E9	232-233	SECTOR NUMBER	0000-0079	00-121	ADDRESS OF BUFFER 2
00EA-00EB	234-235	SECTOR NUMBER	0000-007A	00-122	ADDRESS OF BUFFER 2
00EC-00ED	236-237	SECTOR NUMBER	0000-007B	00-123	ADDRESS OF BUFFER 2
00EE-00EF	238-239	SECTOR NUMBER	0000-007C	00-124	ADDRESS OF BUFFER 2
00F0-00F1	240-241	SECTOR NUMBER	0000-007D	00-125	ADDRESS OF BUFFER 2
00F2-00F3	242-243	SECTOR NUMBER	0000-007E	00-126	ADDRESS OF BUFFER 2
00F4-00F5	244-245	SECTOR NUMBER	0000-007F	00-127	ADDRESS OF BUFFER 2
00F6-00F7	246-247	SECTOR NUMBER	0000-0080	00-128	ADDRESS OF BUFFER 2
00F8-00F9	248-249	SECTOR NUMBER	0000-0081	00-129	ADDRESS OF BUFFER 2
00FA-00FB	250-251	SECTOR NUMBER	0000-0082	00-130	ADDRESS OF BUFFER 2
00FC-00FD	252-253	SECTOR NUMBER	0000-0083	00-131	ADDRESS OF BUFFER 2
00FE-00FF	254-255	SECTOR NUMBER	0000-0084	00-132	ADDRESS OF BUFFER 2

# Memory Man

*Explore the intricacies of your 128's memory*

*By David Kehey*

**T**he C128 has many programmable chips within it. There's the VICII chip for 40-column screen output, the SID chip, which is dedicated to sound, and the own CIA chips which control interfacing and interrupts on the C128. These chips are distributed to those found on the C64 - except for the VICII chip, which has one extra register to control the clock speed of the 8502 processor - and has already been covered in a vast amount of detail. But the C128 has two extra chips. The first is the 64-column screen chip, the second is the Memory Management Unit (MMU for short). In this article, we'll be taking a closer look at the MMU.

## C128 Memory

The C64 was very unique of its time simply because the micro processor could access more than 64K of memory, though not all at the same time. It could do this because the 6502 was updated to allow for special methods of 'bank' selection. RAM and ROM were expanded into blocks, and parts of RAM could be mixed with parts of ROM to produce a full 64K of memory, which the 6510 microprocessor would then see as a full 64K of addressable memory.

The C128 also uses this concept, but Commodore decided to add another 64K of RAM and a whole lot more ROM. The 8502 (which is an updated 6510 to allow 3 mhz operation) couldn't cope with this amount of RAM and ROM, so a chip dedicated to managing all this memory was required. Thus the MMU was born. The upshot of this is that the Commodore has 128K of accessible RAM and a vast amount of ROM to provide the operating system, Basic V1.0, and of course the CIA operating system with Basic V2.0. All this is managed by the MMU so that a different variety of mixtures of RAM and ROM can

be produced to make up the full 64K for the 8502 to address. Each one of these mixtures is called a *configuration*.

## Who uses the MMU

This chip is used frequently by the operating system. When running a Basic program, it has to retrieve the actual program from RAM, but the code for Basic which operates on the Basic instruction is stored in ROM.

You may have also used the MMU. The Basic command BANK provides control over the MMU. This command is only used to allow selection of the possible configurations to allow for running machine code programs from Basic or to peek/peek certain memory locations.

## The 8502 registers

Before I start on the MMU registers, I'll briefly mention the 8502 registers used to manipulate what the 8502 addressing sees in the way of ROM and RAM. These registers are found at locations 900 and 901 - location 900 is the data direction register for register 901.

In 64 mode, they operate as standard to a CIA, and much has already been written on this subject. In 128 mode, however, there is a difference.

Bits 0-2 of register 901 are used to tell the VIC chip and the 8502 where to get information.

## Bits 0 & 1

The VIC chip gets the colour information for the screen from RAM at \$D000, which is a part of the I/O block. However, there is another block of RAM which can be used for colour. This means there are two colour RAM blocks. They both reside at locations \$D800 - \$DFFF, and a block is selected using bits 0 and 1 of address 901.

Bit 1 tells the VIC chip which block to use to display the colour - '0' represents RAM block 0, and a '1'

represents RAM block 1. Bit 0 tells the MMU which block will be used by the 8502 when the I/O section is available to the 8502 (more about this later). This allows the updating of one colour block seen by the 8502 while the other colour RAM block is actually being displayed by the VIC chip. It would then be possible to switch the blocks displayed and update the other RAM block.

On power up, the standard block displayed by VIC and seen by the 8502 is ram block 0.

## Bit 2

Bit 2 selects whether the Character information is within the VIC video bank - 1 means that the character information is found at RAM within the video bank, while 0 means that the character information is taken from ROM at \$D000.

More information can be found with the text regarding the VIC chip.

## The MMU registers

The MMU is controlled using several registers which allow a programmer to control which blocks of RAM and ROM are concurrently visible to the 8502, select which micro processor is being used (the C128 has a 250K mode), and a lot more besides.

What follows is a description of each register and its relative use.

## The configuration register

The configuration register is the one that tells the MMU how to make up the 64K to be addressed by the 8502. This register is found at address \$D000 in the I/O memory block, but can also be found at \$FFF0. The reason for this will become clear very soon.

The first thing this register can do is select which bank of RAM will be used - RAM 0-1 (0-1 is possible with memory expansion). The area from \$C000-\$FFFF can be stored in four different ways. It could contain kernel ROM, other types of ROM or just RAM. The area \$B000-\$FFF is also available as a similar sort of fashion.

# Management

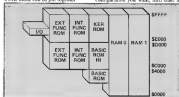
Area \$4000-\$5FFF can either be Basic ROM or RAM.

Lastly, the area \$6000-\$6FFF can either allow us to contain the I/O registers or RAM. This is why \$F000 shadows the MMU register at \$45000. It's possible to switch out the I/O area, but if you did that, how could you then change the 64K configuration again?

Figure 1 shows graphically how the RAM and ROM sit.

Figure 1

The following table summarizes how a 64K block can be put together:



NAME OF AREA	BITS SELECTION
Bank Select	7,6 00 = RAM 0 01 = RAM 1 10 = RAM 2 11 = RAM 3
\$4000-\$5FFF	5,4 00 = Internal ROM 01 = Internal ROM 10 = External ROM 11 = RAM
\$6000-\$6FFF	3,2 00 = BASIC ROM (high) 01 = Internal ROM 10 = External ROM 11 = RAM
\$4000-\$5FFF	1 0 = BASIC ROM (low) 1 = RAM
\$4000-\$6FFF	0 0 = I/O registers 1 1 = RAM + ROM (depending on \$4000-\$6FFF)

is one of the personification registers \$4501-\$4504. Whenever you want that configuration, you just access the corresponding registers \$F000-\$F004. An issue can be any type of store such as STA, STX or STY to that address. By way of example, consider this small program:

```
LDA # $F0
STA $F001 (select configuration $F0)
```

STA \$F001 (select configuration \$F0)

The registers at \$F000-\$F004 must be available in both RAM blocks so as to allow configuration switching anywhere. A further useful point is that when an interrupt occurs, the system could be in any configuration, so at least part of the interrupt service routine must be available in all configurations. Instead of placing the code in 'COMMON RAM', the concept of which is described a little later, code is copied from ROM to RAM in both RAM blocks.

## Mode configuration register

As you know the C128 has three modes of operation, C64, C128 and Z80 mode. Via this register these modes can be selected.

Below is a description of each bit of the register and its function. This register can be found at \$6003, and isn't shadowed.

BIT	DESCRIPTION
0	This selects the processor to be used: 0 = Z80, 1 = 6502
1,2	Not used
3	This bit is called the FWDIR bit, and is used to control the flow of data along the CIA used for serial interfacing. An 0 indicates the fast serial port is for input, and a 1 means that the fast serial port is for output.
4,5	These are used for cartridge port. They can detect signals on the cartridge port and act accordingly. For instance, a C64 cartridge pulls these lines low. The C128 detects this on power up and puts the computer into C64 mode so allow the cartridge software to run.

- 6 - Selects C64 or C128 mode: 0 = C128 mode, 1 = C64  
 7 - 40/38 display key sense: 1 = key depressed

Figure 2.



## Ram Configuration register

Although on a standard C128 there are two separate blocks of 64K RAM, it's possible to have "Common" RAM. Common RAM is RAM which appears in RAM 0 and RAM 1. To illustrate this, try the following example:

```
RANK 1 POKE 254,144
RANK 0 POKE 254,175
RANK 1 PRINT PEEK(254)
```

The result will be 175. This means that a certain part of memory in the 64000 is common which RAM block you are in. You change it while in RAM 0 and switch to RAM 1, but at that certain part of memory the 1502 still sees RAM 0. This register is the one that allows the programmer to control which parts of memory are common, and how much of it is common. The register is found at \$4546.

Figure 2 gives a graphic illustration of the concept of "common" RAM. From "T" to \$FFFF, the contents of the memory seen by the 1502 will always be the same no matter what ram block the 1502 is "seeing". Similarly, from \$0000 to "B" the same applies. Via this register you can move "T" and "B" around.

BITS	DESCRIPTION
1-0	Determine the amount of common Ram
00 =	1K common RAM
01 =	4K common RAM
10 =	32K common RAM
11 =	16K common RAM

- 3-2 These determine which areas of memory will be common. It can be 1K from the top of memory down, 1K from the bottom of memory upwards, or both.  
 1K = the amount specified to be common (explained above)  
 00 = No common RAM  
 01 = Bottom of RAM is common  
 10 = Top of RAM is common  
 11 = Both top and bottom are common  
 3-4 Not used  
 4 VIC chip bank (see below)  
 7 Not used

The VIC chip can be manipulated to use different parts of memory. This bit allows RAM block selection as well: 0 = use RAM block 0; 1 = Use RAM block 1

## Examples

4K common RAM with bottom of RAM only being common (\$05) -

B = \$0FFF (inclusive)

T = not used

common \$0000 - \$05FF

1K common RAM with the top and bottom being common (\$0C) -

B = \$0FFF (inclusive)

T = \$FC00 (exclusive)

16 common \$0000 - \$01FF, \$4000 - \$7FFF

## Page 0 & 1 pointers

Another feature of the MMU is the ability to move page 0 and page 1 to any part of memory. The 1502 is transparent to this, and so the code isn't affected.

For example, suppose the MMU has "moved" page zero to location \$1800. Then, the example -

```
LDA = $80
STA $68
LDA = $40
STA $69
LDY = 0
LDA ($68),Y
```

will actually put 80 40 into locations \$1800 and \$1801 respectively. The final line will of course operate as normal, placing in A the contents of address \$4000.

Whenever the microprocessor accesses page 0, the MMU intercepts this and creates a new address based on the contents of the page 0 pointer. The microprocessor then uses this address thinking it's page zero. What this means is that you can move page zero to another free area if you've no space memory in usual page zero. The microprocessor doesn't recognize any difference, and so all usual page zero 1502 instructions work. A similar description could be given for PAGE 1 pointer as it works in the same way.

Two registers are used to move the page 0 boundary, and two are required for page 1. The two sets of registers work in exactly the same way, so by way of example I shall just talk about the PAGE 0 pointer. Within the 64K memory are page boundaries. There are 256 page boundaries in 64K, which come at every 256 bytes. The table below outlines the page boundaries

```
$0000 - First boundary
$0100 - second boundary
$0200 - Third boundary
$0300 - 256 boundary
```

```
$1000 - boundary
$1000 - first boundary
```

The page 0 must be placed so that it starts at one of these boundaries. The address \$1007 allows you to select 1 of the 256 boundaries you want it to go to.

You can also select which block of RAM you want to place the page you're moving. This is controlled by the register at \$1008. Bits 6-3 are used to select the RAM block, but on a standard 128 the only bit that has any effect is bit 0. An '0' indicates RAM 0, and a '1' selects RAM block 1.

One further thing to note to change the contents of these registers



requires a special operation. The registers are not changed until a write operation is performed on the lower register that controls page boundaries. For page 0 this would be the register at address \$D967. You first write to the bank register at \$D968 (for page 0 [the value is stored but not in the register]). When you write to the boundary register (at \$D967 for page 0), the register is updated and then the bank register is updated with the value stored. For example, moving page 0 to location \$1000 in bank 1 could be done as follows:

```
LDA      set bank 1
$B1
sta      must be done before the
$D968    $D968 write.
LDA      boundary for address $1000
$B0      (or high byte of address)
STA
$D967
```

This isn't a very practical example - if you try this it will crash the computer because the operating system requests access to the correct page zero. A better demonstration is this short program:

```
SEI      stop any interrupts which
may use the stack.
LDA
$B0
STA
$D968
LDA
$B4
STA      set page 1 to address $0400
$D969
LDA
$E4
PHA
PHA
PHA
PHA
PHA
PHA
PHA
LDA
$B1
STA      restore original stack.
$D969
CLI
RRE
```

When you run this, some characters will appear on the 40 column screen.

Note that the stack position isn't preserved by this program, so if you run it again, characters will appear in different places.

To summarize the registers -

\$D967 - Boundary register for page 0  
 \$D968 - Bank register for page 0  
 \$D969 - Boundary register for page 1  
 \$D96A - Bank register for page 1

## MMU version Register

This register is found at address \$D96B, and has two parts to it. The high nibble - i.e. bits 4-7 - contains the number of 64K RAM blocks that the MMU can 'see'. In the case of the standard C128, this is two blocks. The low nibble (bits 0-3) contains the version number of the MMU chip. The current value is 0, so for a standard C128, the contents of this register are \$D9.

## Accessing memory blocks

OK, we've seen the registers that are available. The next thing to consider is how we can access memory currently not seen by the CPU, without appearing to have to change the configuration and run the risk of crashing because we switched out the memory the program was actually running in. This consideration is only required for machine code programmers. Basic contains the BANK command which, although not allowing all the configurations possible, does allow the

ability to access any part of the range of Commodore memory at any time.

The Commodore operating system provides five routines for just this situation. These are -

LDAFAR  
 STAFAR  
 JMFAR  
 JMPFAR  
 CMFAR

One way to call these routines is via the kernel calls, which don't actually call these routines directly but go into kernel ROM first. I don't recommend this, so as to restrict you to having the kernel routines visible to the CPU before you can access these routines. I found on one occasion, that I couldn't get access to KERNAL ROM but required these routines.

The first question you may ask yourself is how can these routines be available from any configuration? The answer lies in how Commodore initializes the MMU on startup. It defines a common area from \$000 to \$03FF, and places the routines within this block. Therefore, no matter which configuration you are in, these routines are always there and so only RAM is available at the address range \$0000 - \$03FF, these routines will always be available.

## LDAFAR, STAFAR & CMFAR

These routines provide extensions to the LDA (\$xx), STA (\$xx),y and CMP (\$xx),y RMC commands. The format for calling is as follows:-

<b>LDAFAR:</b>	lda aa	addr page address used 'aa' for LDAFAR (\$xx),y
	sta \$020a	
	lda config	config is the configuration required
	pr \$02a2	
<b>STAFAR:</b>	lda xx	
	sta \$020a	
	lda config	
	pr \$02a2	
<b>CMFAR:</b>	lda aa	
	sta \$020a	
	lda config	
	pr \$020e	

The kernel version is similar, but it allows the BANK number to be given rather than the configuration. It then converts the bank number to the required configuration, updates the routine to use the correct zero page address, then calls the LDAFAR or STAFAR routines. The call to these routines are as follows (making sure you have the correct configuration) -

```
LDAFAR: ldx #n
        ldx bank
        jr $074

STAFAR: ldx #n
        stx $0269
        ldx bank
        jr $077

JMPFAR: ldx #n
        rdx #0268
        ldx bank
        jr $076
```

## JSEFAR & JMPFAR

These routines provide an extension to the standard JMP and JSR commands found in BASIC. The kernel routines don't have any extra code before they call the routines stored in \$0000 - \$0400, unlike the STAFAR and LDAFAR routines, so the use is basically the same. These routines make use of addresses \$02 - \$09 in zero page to pass all the possible required information internal to the CPU, such as the registers. The memory has the following layout -

```
$02 - Bank number.
$03 - High byte of address to enter
      JMP or JSR
$04 - Low byte part
$05 - Processor status
$06 - Accumulator
$07 - X index register.
$08 - Y index register
$09 - Stack Pointer.
```

We see that the address contained in locations \$03 & \$04 are in a different order to what would normally be expected, as for example to jump to address \$4567 then set -

```
$03 = $45
$04 = $67
```

Apart from location \$08, the JSEFAR

and JMPFAR require all the above information to be setup. Set up as above then -  
JMPFAR jr \$070 or \$02c3  
JSEFAR jr \$006 or \$00d  
(Both calls do exactly the same action.)

On the return after a JSEFAR routine, the values in addresses \$05-\$09 will contain relevant information about the state of the internal registers upon exit of the subroutine called. The information follows the layout described above.

JSEFAR code		
\$026d 20 4563	jr \$02c3	call JMPFAR routine
\$026d 85 06	sta \$06	Save returned A, X, Y
\$026d 86 07	sta \$07	
\$026d 84 08	sty \$08	
\$026d 05	plp	Save Processor status
\$026d 68	pla	
\$026d 85 05	sta \$05	
\$026d 0a	tax	
\$026d 86 09	sta \$09	Save the stack pointer
\$026d e9 00	lda \$00	at configuration
\$026d 80 08 17	lda \$080	In this case BANK 15
\$026d 60	rti	return to caller

JMPFAR		
\$0270 42 00	ldx \$00	Place the address and
\$0270 85 01	ldx \$01	Processor status on the stack
\$0270 48	pha	
\$0270 88	sta	
\$0270 e0 03	sta \$03	
\$0270 90 08	lda \$008	Get Bank
\$0270 26 02	ldx \$02	work out configuration
\$0270 20 6b 01	jr \$00b	
\$0270 84 00 17	lda \$000	Get the values of A, X, Y
\$0270 c5 06	ldx \$06	
\$0270 26 07	ldy \$07	
\$0270 24 08	ldy \$08	
\$0270 60	rti	return to address on stack

These routines would have been very useful if Commodore hadn't made one mistake. Even if you call these routines in the common area and not via the KERNEL, jump, the code then tries to call a kernel routine to convert the bank number to the configuration value. This means to use these routines you must have a configuration where the kernel is visible to the CPU. It also restricts you to only using 16 possible configurations.

An example of this problem can be seen when writing a machine code program which will sit in RAM 1 at high storage key \$F000. This situation

may occur when you want to add a modification to Basic, and you want to sit the program above Basic variable storage which is in RAM 1. If at any time the program needs to call a routine up in the kernel, it can't!

To try the solve the problem, we need to examine the code more closely. The subsequence call to JMPFAR means that when a return is encountered in the code called, it returns back

to the address \$2308, which is common memory.

By looking at the code for these routines given above, we see that the bank conversion call done by both JSEFAR and JMPFAR is actually done by the same piece of code (there's no point looking at the KERNEL routines as all they do is directly jump to the above routines). Also we see that the JSEFAR routine assumes you want to return to configuration 00, but this isn't always the case. It's obvious that to make these routines more flexible, they need to be modified. The problem is that if they are modified,

will then remain compatible with other codes that also call these routines?

## Solution 1

This is just a direct patch on the existing code:

902d from 30 to 2c  
902f from fd to 8c

So the new JMPFAR code is

902c a2 80	ld r, 800	Place the address that
902d 98 01	lds 902c	Processor status on the stack.
902e 7a 01	pha	
902f 05	irs	
9030 08 01	lgr, 501	
9031 80 08	lsc 902c	
9032 a0 02	ld r, 502	Get Bank
9033 30 06 0	ld 5006	Perform an relevant action
9034 8a 00 0	wr 5000	Save configuration
9035 a5 06	lds 506	Get the value of A % Y
9036 a6 07	ld r, 507	
9037 a4 08	ldh 508	
9038 68	rr	return to address on stack with the processor status

You will have to put the contents of location 5000 into 903d before you call the routine. This provides the configuration you want the routine to return to after the JSRFAIR.

If you change the code in this manner then you must specify the configuration in location 902 before calling. Because of this it is no longer compatible with the original routine, and any other call to this routine expecting the usual code is liable to crash. The interrupt routines don't use this part of the code so there isn't any problem. If, however, you are calling Basic or KERNAL routines they may use these routines and it won't work in this modified form. An alternative solution could be

## Solution 2

902d PHP  
902E PHA Save registers that could be affected  
902F LDA Store current configuration  
9030 STA  
9031 IFE  
9032 LDA Store New configuration  
9033 IFE  
9034 STA Patched by calling

9035  
9036 PLA  
9037 PLP  
9038 JSR  
9039  
903A PHP  
903B PHA  
903C LDA  
903D IFE  
903E STA  
903F  
9040 PLA  
9041 PLP  
9042 PHP  
9043 RTS  
return  
Matrix registers  
Call routine (address to be patched)  
Return of routine called  
Save possible affected registers  
get back old configuration  
Restore registers  
return

Place the address that Processor status on the stack.

Get Bank  
Perform an relevant action  
Save configuration  
Get the value of A % Y

return to address on stack with the processor status

This routine mimics a JSRFAIR call. To use it, the following information must be set up:

902C configuration is JSR to  
903D low address of routine to call  
9034 high address of routine to call

For a JMPFAR routine, the code becomes:

902d PHP  
902E PHA  
902F LDA  
9030 IFE  
9031 PLA  
9032 PLP  
9033 JMP  
9034  
9035  
- configuration to be patched by caller  
- address to be patched by caller

The routine here requires that  
902F - configuration  
903E - low address to jump to  
903C - high address to jump to

Calling this routine won't affect any of the registers, and the same rule

applies on return in the case of the JSRFAIR routine. The disadvantage of this system is that code is overwritten from address 902F-903C when the JSRFAIR routine is stored. The code removed is a DMA routine which is used for initializing external memory access. If ever a need for the JSRFAIR code was required, you could patch it in, execute it, then patch back the DMACALL routine either by saving a first using a program, or by just knowing what it is and putting it back. This code isn't used in the normal running of the system. You should obviously select a method that's suited to your needs.

One final thing to note. These routines stored in the area 50000 - 500FF are critical for the programmer to be able to access other blocks of storage.

## Practical uses

We've now covered what registers are available and what software is available for use. But what can you do with it?

With the upgrade to the system software, you can now design programs to run in any RAM block assuming any part of the available Commodore memory. Before, you may have thought you were restricted to just placing programs at certain points because you required access to certain ROMs. You designed programs which require access to RAM 0, but couldn't actually be in RAM 0 in case of corruption of the data stored there. An example of this is the LABEL-LINEIR featured recently in *Four Commodore*. This used the available routines of LDAFAIR and STAFAR, along with the modified JSRFAIR and JMPFAR of solution 1.

## Other possibilities

With so much control over the memory, all sorts of new techniques can be used, and so new programs can be created. The ability to move the location of page zero and page 1 should also broaden your computing horizons considerably. You could move the page 1 and store information in memory via the PHA, much faster than the STA command ever could.

One possibility is the concept of maintaining machine code programs making use of the ability to move page 0 and page 1.

# 128 Corner

*Our regular news letter and general comment spot for  
C128 owners*

**W**elcome to the latest installment of C128 corner. This is the page where C128 owners can find news of new products, general product news and of course letters, comments and general queries from other readers. Don't forget, if YOU have a query, a lot of news or would just like to make a general comment about the C128 then please do write us. If you don't, then C128 Corner will be inevitably bound to produce, so help us to help you by writing us. First a letter

## Resetting the C128

In a recent review of the Super Snapshot cartridge (Your Commodore April 1989), the author noted that there was no switch to turn it off. So to switch his C128 from 64 to 128 mode, he had to switch off his machine and remove the cartridge. This is also the case with the Action Replay cartridge and I imagine many others, much to the annoyance of C128 owners everywhere.

While it is technically possible to add a switch that would switch on a cartridge on the C128, without extra circuitry to protect the computer (switching the switch with the power on could damage the computer). Perhaps anything that adds a few

pounds to the cost of the cartridge partly for the benefit of C128 owners, isn't considered viable by the manufacturer?

I saw the Data! advert for their motherboard - 'switch was my idea', it says. Ok, I thought, just what I need for my C128 Action Replay. But, after buying this, I discovered that while it does switch from one cartridge to another, it does it by switching the 5V power supply to the cartridge on and off. It doesn't allow you to go from C64 mode to C128 mode, even with the board switched to an empty slot.

After a period of switching off, unplugging the cartridge and switching on everytime I used C128 mode, I investigated both cartridge and C64 manual. I found that it's best to turn off any

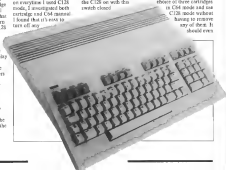
cartridge by simply adding a second switch to the motherboard.

Pin 8, marked ENROM in the C64 manual is the one that locks the 128 into 64 mode. Inserting a switch into this track, on the motherboard would make it possible to switch out the motherboard and cartridge. On Data!'s board, the track from pin 9 continuously goes to the underside of the board for a short distance. At that point I scraped away part of the track, soldered two small wires to each end of the cut track, and connected them to a small switch mounted beside the board's reset button - making pin 9 switchable. Now switching the C128 on with the switch closed

allows the use of cartridges on C64 mode as normal. But turning the computer on with the switch open means it powers up in C128 mode without having to remove any cartridges?

Data!'s board has some protection fitted to protect the computer cartridge when switching between slots, and this may mean that turning the computer off when switching between modes isn't strictly necessary. I haven't had any problems anyway.

Switching as this way may not be in any of the manuals, but it is cheap, requires only minimal soldering and has been used by me for a few months, letting me have the choice of three cartridges in C64 mode and one C128 mode without having to remove any of them. It should even



be possible to add a remote to a cartridge connected across a cut track from PIN 9, though in this case you should turn off the computer before moving the switch.

Pin 9 is located on the user port, and in the 9th from the left on the top row looking from the front of the computer - check your C64 manual for more information.

**Mike Yano, Dagobah**

*It does seem a great idea to modify your motherboard and cartridges in this way. If you were really sophisticated, you could build the switch to pin 9 inside your C128 and put the switch inside the computer. However, be warned - we have been unable to try any of the above alterations out, and don't forget modifying any area as indicated could invalidate any warranty.*

**Make the modifications at your own risk.**

*If you see the list similar modifications that can be made to C128s, please let us know so that we can pass the information on.*

*On the subject of cartridges, you may like to hear about the only cartridge we know of that's available for use in C128 mode. The cartridge is the Warp Speed cartridge, available from those Great of everything to do with the C128 - Financial Systems Software. The cartridge offers turbo disk loading, a machine code monitor and a host of other features. Contact FSEL on (0443) 33433 for more details.*

## C128 In The PD

C128 owners often complain about the lack of software available for their computer. Most of the

software buyers probably haven't even heard of this wonderful machine. If you think that there aren't much C18 software, think again - there's loads of it in the public domain.

Public Domain software is a collection of programs written by various authors with no desire to make money from their labours. The software can be copied and given to other C128 owners, as long as any information that the program's author wishes to be passed on with the program is.

We have recently been contacted by Kingsway Computer Services with details of their C18 public domain library. Prices are extremely cheap, and there are plenty of disks containing full of available programs. For a free catalogue contact

Kingsway Computer Services  
140 Rochdale Road  
Sheffield  
S3 9QE  
Tel: (0344) 988429

## 128 into Amiga will go

Last news just as indicated that a program is now available from FSSS that allows files to be transferred to and from the Commodore Amiga, Atari ST and Apple Macintosh to a C128D or C128 with 1271 disk drive. The package costs £19.95. Please note that this is a file transfer utility only - it does not mean that programs from the aforementioned machines can be used on your C128 or vice versa. The main use for such a utility would be for transferring wordprocessors text files.

## THINKING OF BUYING A NEW CAR? DON'T!

.....without consulting the

## "Handy Little Booklet"

"Handy Little Booklet" is the essential reference book for everyone buying a new car

It lists the main car manufacturers alphabetically, together with each car's current retail price, insurance group, engine size, urban mpg, unleaded competitiveness, Service/Maintenance/Repair costs - based on 50,000 mileage - and tax bracket - for company car drivers.

So before you throw thousands of pounds in a new car basket, just £2 on the "Handy Little Booklet" to help you make the right choice.

To get your copy send £2 plus 25p pence 1 for postage and packaging to: Ron Clary, Company Car, International Trade Publications Ltd., Queensway House, 2 Queensway, Redhill, Surrey RH1 1QS, or telephone car on (0783) 768813. For further details, Ringmaster 320663.



WIN WITH  
**COURSEMASTER**

NEW

## THE COMPUTER HORSE RACING PROGRAMME

- ENTER ANY TIME IN RECORDS - ANY DAILY NEWSPAPER IS ALL YOU NEED
- NEVER out of date - both instant and flip - fast data entry
- ANALYSIS AND COURSE - Now you CAN WIN! THE BOOKING
- Notes on the simple principle that **BEST HORSES BEAT THE ODDS!**
- Clearly illustrated text, excellent in every way plus three **EXCLUSIVE Features** -
- First Second and Third choice sheets for forecasts and Trebles etc
- Recommended bettable type (1 bet)
- A little more out your WORTHED on most popular bets including **WHEELS AND DOUBLES** - see what each says! **PAIRED ODDS** **CARABAS** **HINDS** etc - **Score** **LOST** **WET** and **LONG** **ODDS** bets clearly shown
- **WINNING** bet your **BETTING** slip for you
- No fee to a **BANK ACCOUNT** - GET US **PROFESSIONAL** for potential of your bet in any number of accounts. Keep a complete record of your betting on computer **COURSEMASTER** against your favourite system
- **FLIP** - THE **ANALYSIS COURSEMASTER SYSTEM** - The expert betting system is included in the programme. A system on all regularly published flip cards from month editions. Try it! Don't pay for half money! (this card set has 1st and 2nd)
- **Support** with 20 pages **BETTING** (2, 3, 4) and **MANUAL**

**FREE SET UP OF THE MONTH SENT TO EVERY PURCHASER**  
 (Supported on TAPES at £4.95 for -  
 All domestic Commodore 64  
 128 Amstrad CPCs 128C &  
 1600m also available for  
 popular 64 on Macintosh at  
 £13.95

Please state machine and disc size

**INTRAGET LTD. (UNIT 70, FREEPORT (No Stamp needed) 8 Gifford's Close, Borne Court, Washwood, Warrington, Cheshire, WA3 2SR. Or large SAE for further details.**

# Software for Sale

*If you think that one of our programs looks very interesting, but you can't afford the time to type it in, then our software service will help you out*

**I**t's three o'clock in the morning. You sit at the computer keyboard having just finished a marathon typing session. Lamenting one of the superb programs from Your Commodore, Your fingers reach for the keyboard and press the letters R, U and N. You press RETURN, or back and nothing happens.

Everyone has probably faced this problem. When it does happen it's a matter of spending hours searching through the program for any typing mistakes. No matter how long you look or how many people help you, you can usually guarantee that at least one holey but slips through unnoticed.

The Your Commodore Software Service makes available all of the programs from each issue on both cassette and disk at a price of \$9.00 for disk and \$4.00 for cassette. None of the documentation for the programs is supplied with the software since it is all available in the relevant magazine. Should you not have the magazine then back issues are available from the following address:

INFONET LTD, 5 River Park Estate, Barkswood, Warrs,  
HP4 1BL.  
Tel: (0407) 76661

Please contact this address for price and availability

## The Disk

Programs on the disk will also be supplied as totally working versions, i.e. when possible we will not use Basic Loaders thus making use of the programs much easier. Unfortunately at the moment we cannot duplicate C64 and Plus/4 versions. However programs for these machines will be available on the disk.

What programs are available?

At the top of each article you will find a strip or strips of the article type, C64 Program etc. So that you can see which programs are available on which format, you will also find a couple of symbols after this strip. The symbols have the following meaning:



This symbol means that the program is available on cassette



These programs are available on disk

## Please Note

Since the programs supplied on cassette are total working versions of the program, we do not put disk-only programs on tape. There is no sense in placing a program that expects to be loading from disk on to tape.

## JANUARY 1985

**PREFAB SPRITES** - A powerful sprite editor for the C64

**DAZPRO** - A simple but helpful text processor for the C64. Available on disk and cassette but will only store files on tape.

**UDG COMPRESSOR** - Save on memory when using UDG's in your programs. For C64 only.

**WILLIAM TELL** - Our popular arcade game for the C64 - 4 AUTOMUN - Improve tape loading on your Plus/4 cassette. Only available on disk.

**MINIBASE** - A database for C128 owners

## ORDER CODE

**DISK YDJAN85 £4.00**

**TAPE YCIJAN85 £4.00**

## FEBRUARY 1985

**TAPE MENU** - Add a menu system to your program cassette (C64)

**SOUND EFFECTS** - A superb sound editor for the C64

**F DUMP** - Dump your C64 test screens to printer with ease

**DATA LOADER** - A simple way to enter those rooms of C64 DATA files

**SPRITE LIBRARY** - A collection of birds in your growing library (C64)

**PLAY THE GAME** - A superb fruit machine program for the Plus 4 (Available on disk only)

## ORDER CODE

**DISK YDFEB85 £4.00**

**TAPE YCFEB85 £4.00**

**MARCH 1989**

**PERSONAL FILE** - A cross between a wordprocessor and a database that allows you to set up "cards" that can be quickly stored (C64 Disk only)

**LETTER WRITER** - An 80 column text editor for writing those personal letters (C64)

**BASIC WORKSHOP** - A single key entry system, just like a typewriter (C64)

**HEAD FOR HOME** - Our version of a popular board game for C16 and Plus/4 owners - available on disk only

**SPRITE LIBRARY** - Geometric shapes from this month's installation (C64)

**ELECTRONIC NOTEBOOK** - A personal diary on disk (C64 disk only)

**WILLIAM TELL** - Our very popular arcade adventure for the C64

**ORDER CODE**

**DISK YDMAR89 £6.00**

**TAPE YCMAR89 £4.00**

**APRIL 1989**

**BASEX** - Give your C64 new sound, graphics and roofies commands as well as a machine code assembler with the Basic extension.

**AUTOScroll** - Professional text screen scrolling with this C64 utility.

**BALANCE SHEET** - Keep your bank manager happy by keeping better track of your money with this C64 program (disk only)

**ICON 44** - Add new control to your own programs with this C64 utility.

**80 COLUMNS** - A suite of 80-column graphics commands for C128 80 column users

**FEUS** - A disk file copier for Plus/4 users. Available on disk only

**MULTI-PRECISION ARITHMETIC** - A suite of powerful mathematical routines expanding your C64's number crunching abilities

**STORAGE SPACE** - Use the RAM behind the C64's kernel and Basic ROMs for data storage (available on disk only)

**ORDER CODE**

**DISK YDAF89 £6.00**

**TAPE YCAF89 £4.00**

**MAY 1989**

**ANTI-FREEZE** - Protect your C64 programs from cartridge-based lockouts

**MULTI-COLOURED LIST** - Enhance up your C64 Basic listings by adding on-screen colour

**PRINTFX** - A suite of powerful screen printing commands to add to your Plus/4's Basic. (disk only)

**BANKER 118** - Money Management for C128 disk users

**TURBO TOUCH** - A superb typing trainer for C64 disk users.

**ORDER CODE**

**DISK YDMA89 £6.00**

**TAPE YCMA89 £4.00**

Cartouches or disks are available from March 1989. Please ring the editorial office (01-437 0626) for details of these

**ORDER FORM - PLEASE COMPLETE IN BLOCK CAPITALS**

NAME	QTY	TAPE/DISK	ORDER CODE	PRICE
AUGUST '89		TAPE (£4.00)	YC AUG '89	
AUGUST '89		DISK (£6.00)	YD AUG '89	
OVERSEAS POST £1				
			TOTAL	

NAME

ADDRESS

POST CODE

I enclose a cheque/postal order for £... made payable to ARGUS SPECIALIST PUBLICATIONS LTD.

All orders should be sent to: FOUR COMMODORE, READERS SERVICES, ARGUS HOUSE, BOUNDARY WAY, NEMEL HEMPSTEAD, HERTS HP17 5ST

Please allow 21 days for delivery.

1	1.10	48
2	1.00	DATA 57 173.0 238 41 1
3	0.00	13
35	1.00	DATA 575 0 238 41 14 24
4	0.0	185
35	1.00	DATA 0 50 185 1 50 185
5	0.00	2
35	1.00	DATA 132 2 130.67 18 34
6	0.00	13
35	1.00	DATA 0 238.40 18 34 1
7	0.0	185
35	1.00	DATA 50.185 1 50 173 1
8	0.00	2
35	1.00	DATA 0 130 41 18 34 1
9	0.00	130
35	1.00	DATA 0 141 1.0 238 173 1
10	0.00	141
35	1.00	DATA 3.78 130 130.32 13
11	0.00	13
35	1.00	DATA 93 130 142 83 238
12	0.0	185
35	1.00	DATA 141 85 234 142 85
13	0.00	2
35	1.00	DATA 234 238 13 185 1.1
14	0.0	185
35	1.00	DATA 140 13 234 76 0 23
15	0.00	13
35	1.00	DATA 234 133 83 234 233
16	0.00	2
35	1.00	DATA 141 143 13 234 2
17	0.0	185
35	1.00	DATA 133 14 234 231 5.4
18	0.00	14
35	1.00	DATA 0 143 18 234 173 5
19	0.00	2
35	1.00	DATA 50 234 173 62 234
20	0.0	185
35	1.00	DATA 173 64 234 141 50
21	0.00	2
35	1.00	DATA 234 141 64 234 173
22	0.00	13
35	1.00	DATA 140 73 134 76 3.2
23	0.0	185
35	1.00	DATA 142 82 238 173
24	0.00	2
35	1.00	DATA 93 141 141 13 234
25	0.00	13
35	1.00	DATA 140 84 84 234 141
26	0.00	2
35	1.00	DATA 84 234 140 84 234
27	0.00	13
35	1.00	DATA 76 140 234 173 85
28	0.00	2
35	1.00	DATA 234 85 141 233 140
29	0.0	185
35	1.00	DATA 141 234 142 83 234
30	0.00	2
35	1.00	DATA 173 31 234 30 243
31	0.00	2
35	1.00	DATA 234 141 94 234 142
32	0.00	2
35	1.00	DATA 173 50 234 184 234
33	0.00	13
35	1.00	DATA 140 173 83 234 185
34	0.00	2
35	1.00	DATA 141 141
35	1.00	DATA 130 234 173 185 13
36	0.00	13
35	1.00	DATA 141 141 234 140
37	0.00	2
35	1.00	DATA 141 141 185 234 140
38	0.00	2
35	1.00	DATA 141 141 185 234 140
39	0.00	2
35	1.00	DATA 141 141 185 234 140

61	1440	BATH	227	94	224	241	148	
	324	173	218					
62	1440	BATH	224	227	94	224	141	
	124	224	224					
63	1440	BATH	124	224	14	14	148	
	8	141	148					
64	1440	BATH	224	241	148	224	74	
	124	227	224					
65	1440	BATH	8	241	93	224	173	2
	2	224	141					
66	1440	BATH	93	224	14	82	224	4
	8	93	224					
67	1440	BATH	14	93	224	48	83	27
	4	14	93					
68	1440	BATH	124	48	93	224	173	
	82	224	248					
69	1440	BATH	3	224	93	224	224	2
	2	224	141					
70	1440	BATH	174	224	148	93	224	
	48	22	224					
71	1440	BATH	8	173	148	224	224	
	93	224	241					
72	1440	BATH	3	173	93	224	141	
	224	224	224					
73	1440	BATH	93	224	141	124	224	
	173	141	224					
74	1440	BATH	141	93	224	173	148	
	224	141	41					
75	1440	BATH	224	74	93	224	218	
	82	224	24					
76	1440	BATH	3	224	174	93	224	
	224	93	224					
77	1440	BATH	224	141	224	141	224	
	224	141	224					
78	1440	BATH	224	141	224	141	224	
	224	141	224					
79	1440	BATH	224	141	224	141	224	
	224	141	224					
80	1440	BATH	224	141	224	141	224	
	224	141	224					
81	1440	BATH	224	141	224	141	224	
	224	141	224					
82	1440	BATH	224	141	224	141	224	
	224	141	224					
83	1440	BATH	224	141	224	141	224	
	224	141	224					
84	1440	BATH	224	141	224	141	224	
	224	141	224					
85	1440	BATH	224	141	224	141	224	
	224	141	224					
86	1440	BATH	224	141	224	141	224	
	224	141	224					
87	1440	BATH	224	141	224	141	224	
	224	141	224					
88	1440	BATH	224	141	224	141	224	
	224	141	224					
89	1440	BATH	224	141	224	141	224	
	224	141	224					
90	1440	BATH	224	141	224	141	224	
	224	141	224					
91	1440	BATH	224	141	224	141	224	
	224	141	224					
92	1440	BATH	224	141	224	141	224	
	224	141	224					
93	1440	BATH	224	141	224	141	224	
	224	141	224					
94	1440	BATH	224	141	224	141	224	
	224	141	224					
95	1440	BATH						

[illegible]



# C128 PROGRAM

19	2099 DATA 100 200 170 0 204	47	2099 DATA 200 200 100 200 10	74	2109 DATA 90 204 70 204 200
20	2100 DATA 200 100 200 100 20	48	2100 DATA 200 200 100 200 20	75	2100 DATA 70 100 170 100 20
21	2101 DATA 70 170 70 204 204	49	2101 DATA 200 200 100 200 20	76	2101 DATA 200 200 100 200 20
22	2102 DATA 204 100 170 70 204 204	50	2102 DATA 200 200 100 200 20	77	2102 DATA 170 20 204 204 20 4
23	2103 DATA 100 170 70 204 204	51	2103 DATA 200 200 100 200 20	78	2103 DATA 70 200 200 20 7 0 7
24	2104 DATA 170 204 200 204 200	52	2104 DATA 200 200 100 200 20	79	2104 DATA 0 0 204 0 10 200 0
25	2105 DATA 100 200 200 200 200 70	53	2105 DATA 200 200 100 200 20	80	2105 DATA 70 100 100 1 200 10
26	2106 DATA 200 200 200 200 200	54	2106 DATA 200 200 100 200 20	81	2106 DATA 100 0 0 0 0 0 0
27	2107 DATA 100 200 200 200 200	55	2107 DATA 200 200 100 200 20	82	2107 DATA 100 0 0 0 0 0 0
28	2108 DATA 200 200 200 200 200	56	2108 DATA 200 200 100 200 20	83	2108 DATA 0 0 0 200 0 0 0 0
29	2109 DATA 200 200 200 200 200	57	2109 DATA 200 200 100 200 20	84	2109 DATA 0 0 0 200 0 0 0 0
30	2110 DATA 200 200 200 200 200	58	2110 DATA 200 200 100 200 20	85	2110 DATA 0 0 0 200 0 0 0 0
31	2111 DATA 200 200 200 200 200	59	2111 DATA 200 200 100 200 20	86	2111 DATA 0 0 0 200 0 0 0 0
32	2112 DATA 200 200 200 200 200	60	2112 DATA 200 200 100 200 20	87	2112 DATA 0 0 0 200 0 0 0 0
33	2113 DATA 200 200 200 200 200	61	2113 DATA 200 200 100 200 20	88	2113 DATA 0 0 0 200 0 0 0 0
34	2114 DATA 200 200 200 200 200	62	2114 DATA 200 200 100 200 20	89	2114 DATA 0 0 0 200 0 0 0 0
35	2115 DATA 200 200 200 200 200	63	2115 DATA 200 200 100 200 20	90	2115 DATA 0 0 0 200 0 0 0 0
36	2116 DATA 200 200 200 200 200	64	2116 DATA 200 200 100 200 20	91	2116 DATA 0 0 0 200 0 0 0 0
37	2117 DATA 200 200 200 200 200	65	2117 DATA 200 200 100 200 20	92	2117 DATA 0 0 0 200 0 0 0 0
38	2118 DATA 200 200 200 200 200	66	2118 DATA 200 200 100 200 20	93	2118 DATA 0 0 0 200 0 0 0 0
39	2119 DATA 200 200 200 200 200	67	2119 DATA 200 200 100 200 20	94	2119 DATA 0 0 0 200 0 0 0 0
40	2120 DATA 200 200 200 200 200	68	2120 DATA 200 200 100 200 20	95	2120 DATA 0 0 0 200 0 0 0 0
41	2121 DATA 200 200 200 200 200	69	2121 DATA 200 200 100 200 20	96	2121 DATA 0 0 0 200 0 0 0 0
42	2122 DATA 200 200 200 200 200	70	2122 DATA 200 200 100 200 20	97	2122 DATA 0 0 0 200 0 0 0 0
43	2123 DATA 200 200 200 200 200	71	2123 DATA 200 200 100 200 20	98	2123 DATA 0 0 0 200 0 0 0 0
44	2124 DATA 200 200 200 200 200	72	2124 DATA 200 200 100 200 20	99	2124 DATA 0 0 0 200 0 0 0 0
45	2125 DATA 200 200 200 200 200	73	2125 DATA 200 200 100 200 20	100	2125 DATA 0 0 0 200 0 0 0 0
46	2126 DATA 200 200 200 200 200	74	2126 DATA 200 200 100 200 20	101	2126 DATA 0 0 0 200 0 0 0 0
47	2127 DATA 200 200 200 200 200	75	2127 DATA 200 200 100 200 20	102	2127 DATA 0 0 0 200 0 0 0 0
48	2128 DATA 200 200 200 200 200	76	2128 DATA 200 200 100 200 20	103	2128 DATA 0 0 0 200 0 0 0 0
49	2129 DATA 200 200 200 200 200	77	2129 DATA 200 200 100 200 20	104	2129 DATA 0 0 0 200 0 0 0 0
50	2130 DATA 200 200 200 200 200	78	2130 DATA 200 200 100 200 20	105	2130 DATA 0 0 0 200 0 0 0 0
51	2131 DATA 200 200 200 200 200	79	2131 DATA 200 200 100 200 20	106	2131 DATA 0 0 0 200 0 0 0 0
52	2132 DATA 200 200 200 200 200	80	2132 DATA 200 200 100 200 20	107	2132 DATA 0 0 0 200 0 0 0 0
53	2133 DATA 200 200 200 200 200	81	2133 DATA 200 200 100 200 20	108	2133 DATA 0 0 0 200 0 0 0 0
54	2134 DATA 200 200 200 200 200	82	2134 DATA 200 200 100 200 20	109	2134 DATA 0 0 0 200 0 0 0 0
55	2135 DATA 200 200 200 200 200	83	2135 DATA 200 200 100 200 20	110	2135 DATA 0 0 0 200 0 0 0 0
56	2136 DATA 200 200 200 200 200	84	2136 DATA 200 200 100 200 20	111	2136 DATA 0 0 0 200 0 0 0 0
57	2137 DATA 200 200 200 200 200	85	2137 DATA 200 200 100 200 20	112	2137 DATA 0 0 0 200 0 0 0 0
58	2138 DATA 200 200 200 200 200	86	2138 DATA 200 200 100 200 20	113	2138 DATA 0 0 0 200 0 0 0 0
59	2139 DATA 200 200 200 200 200	87	2139 DATA 200 200 100 200 20	114	2139 DATA 0 0 0 200 0 0 0 0
60	2140 DATA 200 200 200 200 200	88	2140 DATA 200 200 100 200 20	115	2140 DATA 0 0 0 200 0 0 0 0
61	2141 DATA 200 200 200 200 200	89	2141 DATA 200 200 100 200 20	116	2141 DATA 0 0 0 200 0 0 0 0
62	2142 DATA 200 200 200 200 200	90	2142 DATA 200 200 100 200 20	117	2142 DATA 0 0 0 200 0 0 0 0
63	2143 DATA 200 200 200 200 200	91	2143 DATA 200 200 100 200 20	118	2143 DATA 0 0 0 200 0 0 0 0
64	2144 DATA 200 200 200 200 200	92	2144 DATA 200 200 100 200 20	119	2144 DATA 0 0 0 200 0 0 0 0
65	2145 DATA 200 200 200 200 200	93	2145 DATA 200 200 100 200 20	120	2145 DATA 0 0 0 200 0 0 0 0
66	2146 DATA 200 200 200 200 200	94	2146 DATA 200 200 100 200 20	121	2146 DATA 0 0 0 200 0 0 0 0
67	2147 DATA 200 200 200 200 200	95	2147 DATA 200 200 100 200 20	122	2147 DATA 0 0 0 200 0 0 0 0
68	2148 DATA 200 200 200 200 200	96	2148 DATA 200 200 100 200 20	123	2148 DATA 0 0 0 200 0 0 0 0
69	2149 DATA 200 200 200 200 200	97	2149 DATA 200 200 100 200 20	124	2149 DATA 0 0 0 200 0 0 0 0
70	2150 DATA 200 200 200 200 200	98	2150 DATA 200 200 100 200 20	125	2150 DATA 0 0 0 200 0 0 0 0
71	2151 DATA 200 200 200 200 200	99	2151 DATA 200 200 100 200 20	126	2151 DATA 0 0 0 200 0 0 0 0
72	2152 DATA 200 200 200 200 200	100	2152 DATA 200 200 100 200 20	127	2152 DATA 0 0 0 200 0 0 0 0
73	2153 DATA 200 200 200 200 200	101	2153 DATA 200 200 100 200 20	128	2153 DATA 0 0 0 200 0 0 0 0
74	2154 DATA 200 200 200 200 200	102	2154 DATA 200 200 100 200 20	129	2154 DATA 0 0 0 200 0 0 0 0
75	2155 DATA 200 200 200 200 200	103	2155 DATA 200 200 100 200 20	130	2155 DATA 0 0 0 200 0 0 0 0
76	2156 DATA 200 200 200 200 200	104	2156 DATA 200 200 100 200 20	131	2156 DATA 0 0 0 200 0 0 0 0
77	2157 DATA 200 200 200 200 200	105	2157 DATA 200 200 100 200 20	132	2157 DATA 0 0 0 200 0 0 0 0
78	2158 DATA 200 200 200 200 200	106	2158 DATA 200 200 100 200 20	133	2158 DATA 0 0 0 200 0 0 0 0
79	2159 DATA 200 200 200 200 200	107	2159 DATA 200 200 100 200 20	134	2159 DATA 0 0 0 200 0 0 0 0
80	2160 DATA 200 200 200 200 200	108	2160 DATA 200 200 100 200 20	135	2160 DATA 0 0 0 200 0 0 0 0
81	2161 DATA 200 200 200 200 200	109	2161 DATA 200 200 100 200 20	136	2161 DATA 0 0 0 200 0 0 0 0
82	2162 DATA 200 200 200 200 200	110	2162 DATA 200 200 100 200 20	137	2162 DATA 0 0 0 200 0 0 0 0
83	2163 DATA 200 200 200 200 200	111	2163 DATA 200 200 100 200 20	138	2163 DATA 0 0 0 200 0 0 0 0
84	2164 DATA 200 200 200 200 200	112	2164 DATA 200 200 100 200 20	139	2164 DATA 0 0 0 200 0 0 0 0
85	2165 DATA 200 200 200 200 200	113	2165 DATA 200 200 100 200 20	140	2165 DATA 0 0 0 200 0 0 0 0
86	2166 DATA 200 200 200 200 200	114	2166 DATA 200 200 100 200 20	141	2166 DATA 0 0 0 200 0 0 0 0
87	2167 DATA 200 200 200 200 200	115	2167 DATA 200 200 100 200 20	142	2167 DATA 0 0 0 200 0 0 0 0
88	2168 DATA 200 200 200 200 200	116	2168 DATA 200 200 100 200 20	143	2168 DATA 0 0 0 200 0 0 0 0
89	2169 DATA 200 200 200 200 200	117	2169 DATA 200 200 100 200 20	144	2169 DATA 0 0 0 200 0 0 0 0
90	2170 DATA 200 200 200 200 200	118	2170 DATA 200 200 100 200 20	145	2170 DATA 0 0 0 200 0 0 0 0
91	2171 DATA 200 200 200 200 200	119	2171 DATA 200 200 100 200 20	146	2171 DATA 0 0 0 200 0 0 0 0
92	2172 DATA 200 200 200 200 200	120	2172 DATA 200 200 100 200 20	147	2172 DATA 0 0 0 200 0 0 0 0
93	2173 DATA 200 200 200 200 200	121	2173 DATA 200 200 100 200 20	148	2173 DATA 0 0 0 200 0 0 0 0
94	2174 DATA 200 200 200 200 200	122	2174 DATA 200 200 100 200 20	149	2174 DATA 0 0 0 200 0 0 0 0
95	2175 DATA 200 200 200 200 200	123	2175 DATA 200 200 100 200 20	150	2175 DATA 0 0 0 200 0 0 0 0
96	2176 DATA 200 200 200 200 200	124	2176 DATA 200 200 100 200 20	151	2176 DATA 0 0 0 200 0 0 0 0
97	2177 DATA 200 200 200 200 200	125	2177 DATA 200 200 100 200 20	152	2177 DATA 0 0 0 200 0 0 0 0
98	2178 DATA 200 200 200 200 200	126	2178 DATA 200 200 100 200 20	153	2178 DATA 0 0 0 200 0 0 0 0
99	2179 DATA 200 200 200 200 200	127	2179 DATA 200 200 100 200 20	154	2179 DATA 0 0 0 200 0 0 0 0
100	2180 DATA 200 200 200 200 200	128	2180 DATA 200 200 100 200 20	155	2180 DATA 0 0 0 200 0 0 0 0



# MEGALAND

0703 332225

## CREDIT CARD MAIL ORDER

### AMIGA 500 MEGA PACK £399.00 INC

INCLUDING ROMAN WORKBOOKS 1,2, UTILITIES, MANUALS, BASIC TUTORIAL, PLUS FREE TV MODULATOR, JOYSTICK AND ANY TEN FREE GAMES FROM OUR SELECTION BELOW

Compendix, Workbook Quest, Splitting Image, 1st Soccer, Whizz, Dystopian 68, Helibot, Backlash, Dynapac, Phoenix of the Jedi, Eliminator, Commander, Gladiator, Secondary, Quadrangle, Star Game, Powerplay, Montages, Phoenix, Motul

### AMIGA HARDWARE

AMIGA 500 TV MOD	£299 inc
AMIGA 500/500+16	£299 inc
16MB RAM	
16MB COLOUR MONITOR	£299 inc
AMIGA 500/500+16	£299 inc
COLOUR MONITOR	
AMIGA 500/500+16	£299 inc
COLOUR MONITOR	

### A2000 HARDWARE

AMIGA 500	£299 inc
AMIGA 500/500+16	£299 inc
16MB RAM	
16MB COLOUR MONITOR	£299 inc
AMIGA 500/500+16	£299 inc
COLOUR MONITOR	
AMIGA 500/500+16	£299 inc
COLOUR MONITOR	

### PRINTERS

EPSON LQW-8000	£299 inc
EPSON LQW-8000	£299 inc
EPSON LQW-8000	£299 inc
EPSON LQW-8000	£299 inc
EPSON LQW-8000	£299 inc
EPSON LQW-8000	£299 inc
EPSON LQW-8000	£299 inc
EPSON LQW-8000	£299 inc

### AMIGA EXTERNAL DRIVES

AMIGA 500 1.5 MB	£299 inc
AMIGA 500 1.5 MB	£299 inc
AMIGA 500 1.5 MB	£299 inc
AMIGA 500 1.5 MB	£299 inc
AMIGA 500 1.5 MB	£299 inc

### AMIGA ACCESSORIES

AMIGA 500 RAM	£299 inc
AMIGA 500 RAM	£299 inc
AMIGA 500 RAM	£299 inc
AMIGA 500 RAM	£299 inc
AMIGA 500 RAM	£299 inc
AMIGA 500 RAM	£299 inc
AMIGA 500 RAM	£299 inc
AMIGA 500 RAM	£299 inc

### A2000 ACCESSORIES

AMIGA 500 RAM	£299 inc
AMIGA 500 RAM	£299 inc
AMIGA 500 RAM	£299 inc
AMIGA 500 RAM	£299 inc
AMIGA 500 RAM	£299 inc
AMIGA 500 RAM	£299 inc
AMIGA 500 RAM	£299 inc
AMIGA 500 RAM	£299 inc

### COLOUR PRINTERS

EPSON LQW-8000	£299 inc
EPSON LQW-8000	£299 inc
EPSON LQW-8000	£299 inc
EPSON LQW-8000	£299 inc
EPSON LQW-8000	£299 inc

### LASER PRINTERS

EPSON LQW-8000	£299 inc
EPSON LQW-8000	£299 inc
EPSON LQW-8000	£299 inc
EPSON LQW-8000	£299 inc
EPSON LQW-8000	£299 inc

### DRIVES

AMIGA 500 RAM	£299 inc
AMIGA 500 RAM	£299 inc
AMIGA 500 RAM	£299 inc
AMIGA 500 RAM	£299 inc
AMIGA 500 RAM	£299 inc
AMIGA 500 RAM	£299 inc
AMIGA 500 RAM	£299 inc
AMIGA 500 RAM	£299 inc

### MONITORS

AMIGA 500 RAM	£299 inc
AMIGA 500 RAM	£299 inc
AMIGA 500 RAM	£299 inc
AMIGA 500 RAM	£299 inc
AMIGA 500 RAM	£299 inc
AMIGA 500 RAM	£299 inc
AMIGA 500 RAM	£299 inc
AMIGA 500 RAM	£299 inc

### A500/A2000 MONITORS

AMIGA 500 RAM	£299 inc
AMIGA 500 RAM	£299 inc
AMIGA 500 RAM	£299 inc
AMIGA 500 RAM	£299 inc
AMIGA 500 RAM	£299 inc
AMIGA 500 RAM	£299 inc
AMIGA 500 RAM	£299 inc
AMIGA 500 RAM	£299 inc

CREDIT CARD MAIL ORDER  
SPECIAL HOTLINE FOR EXPORT  
TEL 0703 332225

ALL OTHERS SUBJECT TO AVAILABILITY  
OFFERS SUBJECT TO CHANGE IN PRICE  
WITHOUT NOTICE

### MEGA SUPER OFFER

AMIGA 5000  
with 20 MB Hard Drive  
1024x Colour Monitor  
£399.00 inc VAT

### SPECIAL OFFER

Amiga 5000 1.5 MB  
1024x Colour Monitor  
£379 inc VAT

### MEGA SUPER OFFER

AMIGA 5000  
Monitor 1024x  
RAM Expansion  
£399 inc

### SPECIAL OFFERS

EPSON LQW-8000  
24 PIN PRINTER  
£299 inc VAT

WESTERN DIGITAL  
HARD CARD 32MB  
£259.00 inc VAT



ALL PRICES INCLUDE DELIVERY CHARGE

Ep 10-11 Millbrook Road East, Southampton

I wish to order

I also take cheque PO for £

De charge my Account/Visa No

Name

Address

Postcode

Tel No

Signature

Exp date

Inc VAT

MEGALAND





# Mailbag

## Your chance to air your views in Your Commodore

Having very recently acquired an Osborne 1185N, I was particularly interested to read Norman Doyle's articles in various editions of *Your Commodore*. A comment made in that article has me a little worried. The concern the remarks made about floppy disks in the paragraphs on the Commodore 1270/71. I wonder if the frozen damage caused by turning disks over to use "side 2" is a problem with the drives - would it be harmful to my GC 1185N? If so then there is no economy in buying double sided disks.

I would be grateful for your comments on this.  
F.D.A. Roche, London

I found the article "Whisk Drive" by Norman Doyle in the June '88 issue of *Your Commodore* most interesting. With reference to his comments on the 1251:

I have Commodore models C64 with 1241 Disk Drive and a Plus/4 with 1251 Disk Drive. Both the Plus/4 and C64 will accept either type of disk drive since they are fitted with the usual 1-pin DIN sockets for serial data chaining of peripherals and a User Port having a 2x25 contacts interface. The User Port is SOI (the same as that of the C64). The 1251 drive has a fitted data transfer lead which terminates in a custom made cartridge to fit into the User Port. This cartridge has a "through" connection in which to despatch other cartridges, although I have never seen or heard of any Osborne (more 1251) could be used, but I mainly plug the 1241 into the alternative serial socket when I need two drives running. Visually and physically the 1251 is similar to the

1241 except there is no provision for despatching a 4 pin DIN-connected printer, but this is unnecessary in view of the facility being provided on the Plus/4 and C64. The "lock" is of the "turn down tab" similar to the Excelsior and West Chip drives.

Performance comparison was carried out by entering a 34 block (actually 8,512 bytes) data-storage program from the 1251 drive into the Plus/4. A timing program - see below - was entered into the C64 and used to measure the time for the Plus/4 to load the 34 block program. Loading time was consistently nine seconds flat. The disk was then transferred to the 1241 for loading into the C64 and the timing program entered into the Plus/4. The time was repeated on the 1241 and loading took 23 seconds. Using these figures 1251 transfers at 9440 baud compared with 1241 transferring at 1702 baud.

This clearly indicates an improvement of nearly 200% when loading program files via the 1251. Speeds may well differ on other data operations or flag systems.

The Commodore 1251 disk was distributed in small quantities, but rarely, if ever, advertised. I bought mine from Goro's in Nottingham's branch of Debenhams late in 1988 for about £139. They had several dozen for sale at the time. The odd one has been offered in FOR SALE columns of various publications.

Although prices are not my forte one must accept that there were speed appeal, especially to the younger members of society, guaranteed the need for advertising, and is a necessary source of income for journals and magazines. Notwithstanding I would suggest that the more mature enthusiasts will continue to subscribe

for many years if more articles in similar vein to Mr Doyle's are forthcoming.

\* I wish that program  
2 TIS "000000"  
3 GETAS IPAS="THEN 3  
4 PRINT TIS: rem prints time when  
5 RUN rem any key pressed  
Michael Stock, Nottingham

I made a mistake! I sold my 1241 and bought an excelsior!  
Oh, don't get me wrong, it looks good, sounds good and loaded all my games, until I went to load *Defender of the Crown*. Then the running began, crash after crash after crash. Instant told me that the Excelsior was at fault because the game had always loaded on my 1241 with no agree.

I rang Everham Micro to explain the problems and was informed that, "Yes, *Defender of the Crown* would not load on the Excelsior drive" - for a moment I was lost for words! The dry voice on the phone then added "the same applies to *Flying Shark*, they are the only two".

Apparently, Everham have had to give away at least one Amiga computer because of *Defender of the Crown*. Although the manager quickly pointed out how he learned that they had been because the game did tend to run-load successfully, even on the 1241. He went on to suggest the fault lay with the software company, saying that they were at a loss as to the reason for this.

I then stated that I was very unhappy with the situation and was told that if I sent my copy of *Defender* to them (Everham) they would re-configure the software to run on the Excelsior. The chap did explain that the Excelsior could not be made totally compatible with the Commodore because copyright law would be infringed.

Alan Park, Chafford, Essex

**F**ollowing Norman Doyle's article on disk drives in the February issue, the letters have been flooding in. Now's your chance to get a few things straight.

Using the reverse side of a disk known affectionately as a 'floppy' disk, is an established practice, born out of the need to cut costs. Now that floppy disks are cheaply available to the general public, the drawbacks become more obvious.

Because the disk is made to spin backwards when you turn a cover to use the other side, the increased friction from you has the differentials must reduce the rotation speed, but so slightly as not to affect a sensitive drive. The mass drawback, with floppy, is that squaring them backwards means at the last inconveniently wrapped around the disk's outer rim. Although this is not a major nuisance, it tends to make the floppy less reliable. I only use floppies for archiving.

average, just to save space on my disk  
here

The question of drive compatibility has stayed in my mind one or two times already. It's added me to contemplations of a manufacturer claims his product is "Commodore-compatible." It should be put that! There's no way anyone can make a truly compatible drive, one would involve replacing much that is protected by copyright laws! Compatibility, however, can't be a bad thing in this case, the customer will benefit from having a wider choice, after all! Now to mind also, Commodore's history of incompatibility! The 1540 was not totally compatible with the older 1540 drive, the new 1540C is definitely not compatible with the 1540. Even CB's themselves have undergone design changes that have caused compatibility problems! You can't stop progress though, and as long as the software publishers keep ahead of these new developments, everything

Downloaded At: 11:53 11 September 2009

Ernest Moore, distributor for the Excelsior drum, cradled up grooves away two Amgas, and there offer of a conversion job on more unloadable guitar still stands. They are quick to point out that that *Defender Of The Crown* originally would load on the 1541C. The situation was reversed when the game was modified? Ernest also pointed out that the Excelsior loads more games than Commodore's own 1541C? Ya pay ya money and ya take ya choice!

Owing to the lack of popularity of the Ples 4 and C16 computers, Comma-line decided to abandon the 1951 drive some time ago. But?

If you want to write in-AL, address your letters to:

Four Commodore  
Argon Home,  
Boundary Way,  
Hemel Hempstead,  
Herts HP1 1ST

### CLASSIFIED ADVERTISEMENT COUPON

COMMODORE

Write your ad in 8,000 CHARACTERS or less and below. Taking the section you wish it to appear under INCLUDING NAME AND ADDRESS IN TWO WORDS (30 CHARACTERS) and send it to TEAM CONNECTIONS, CLASSIFIED DEPARTMENT, 8000 HOLLY, BOUNSBURY WET, HENL. BUNSBURY, WET 020

Charge - \$47 minimum plus \$100 no compensation for cancellations. All ads must be pre-paid.

 [New York](#)  [California](#)  [Texas](#)  [Florida](#)  [Illinois](#)  [Pennsylvania](#)  [Ohio](#)  [Michigan](#)  [Indiana](#)  [Georgia](#)  [Alabama](#)  [Louisiana](#)  [Mississippi](#)  [Arkansas](#)


**CLASSIFIED COUPON**

ALL CLASSIFIED ADVERTISEMENTS MUST BE PRE-PAY.  
THERE ARE NO REFUNDMENTS OR CANCELLATIONS.

I enclose my Charge/Postal Order for £ \_\_\_\_\_ for \_\_\_\_\_  
 magazine, made payable to Angus & Sonnet Publications  
 (Circle as appropriate) of \_\_\_\_\_

100

1000

**Figure 1**

	Springer
---	----------

© 2005 Blackwell Publishing Ltd *Journal of Internal Medicine* 258: 105–112

Figure 1 illustrates a 1D lattice with 12 sites. The top row shows the initial state with a single electron (black dot) at site 1. The bottom row shows the state after a time step, with the electron moved to site 2. The lattice is represented by a horizontal line with vertical bars at each site. The sites are numbered 1 to 12 from left to right.

10

[illegible]

1000











# Emlyn Hughes



## INTERNATIONAL



**GAME  
OF THE YEAR!**

**WINNER OF CCI OSKARS FOR  
BEST SPORTS SIMULATION  
AND BEST GAME OF THE YEAR!**

*Quite simply the best football  
simulation available for the 64.  
Nothing short of superb.*

**ZZAP 64**

**AVAILABLE NOW FOR  
COMMODORE 64  
£9.95 tape £12.95 disk**  
Coming soon for Spectrum & Amstrad

## Audiogenic

Audiogenic Software Limited, Winchester House,  
Canning Road, Harrow HA1 2JJ, England

Order by phone on 01 861 1166

